

PROGRAMOVACÍ NÁSTROJ

PES-MENU

1. ed. 6. 99

OBSAH

1. CO JE TO PESMENU	4
2. APLIKACE A JEJÍ OVLÁDÁNÍ	5
3. OVLÁDACÍ PRVKY APLIKACE V PESMENU	8
3.1. ZOBRAZOVÁNÍ NA DISPLEJI A JEHO PRINCIP	8
3.2. OVLÁDACÍ PRVEK MENU A MENU S NADPISEM	10
3.3. OVLÁDACÍ PRVEK EDITOR VÝČTU A ODKAZ NA VÝČET	12
3.4. OVLÁDACÍ PRVEK EDITOR HODNOTY A ODKAZ NA HODNOTU	14
3.5. OVLÁDACÍ PRVEK PŘÍKAZ	16
3.6. ZOBRAZOVACÍ PRVEK OBRAZOVKA	16
3.7. ODKAZ NA UŽIVATELSKÝ TYP	17
4. POPIS PROGRAMU PESMENU	18
4.1. ZÁKLADNÍ OBRAZOVKA A PŘÍKAZY MENU	19
4.2. EDITOR MENU	22
4.3. EDITOR ZOBRAZENÍ OVLÁDACÍCH PRVKŮ MENU	26
4.4. EDITOR ZNAKOVÉ SADY	30
4.5. NASTAVENÍ PROSTŘEDÍ PESMENU	31
5. ŘEŠENÉ ÚLOHY OVLÁDACÍCH PANELŮ	33
5.1. RUČKOVÉ MĚŘIDLO NA DISPLEJI AUTOMATU ŘADY PES-K	33
5.2. PŘEPÍNAČ VÝSTUPŮ Y0 AŽ Y7 A EDITOR NAPĚTÍ NA VÝSTUPU O0	40
6. ZÁVĚREM	48

1. CO JE TO PESMENU

PESMENU je programovací nástroj pro generování zdrojových textů ovládní klávesnice a displeje v programovacím jazyce automatů systému PES. Programovací nástroj umožňuje:

- a) Návrh ovládní aplikace pomocí systému nabídek (**MENU**), editorů hodnoty, výčtu a systémem příkazů.
- b) Zobrazování obsahu proměnných systémem odkazů. Odkazy mohou zobrazovat hodnotu číslem, výčtem a v uživatelském formátu.
- c) Tvorbu uživatelských zobrazovacích znaků ve vlastních znakových sadách (vhodné např. pro podporu češtiny apod.).
- d) Prohlížení navrženého zobrazení.
- e) Generování zdrojového textu v programovacím jazyce automatů systému PES.

Oblasti nasazení programovacího nástroje PESMENU

Hlavní předpokládanou oblastí použití programovacího nástroje PESMENU je tvorba ovládacích panelů aplikací. Programovací nástroj je možné použít v zásadě dvěma způsoby: buď jako generátor zdrojového textu pro ovládací panel aplikace a nebo jako generátor zdrojového textu informační obrazovky o stavu aplikace. Programový nástroj je možné použít i pro sjednocení ovládní starších, již hotových aplikací s aplikacemi novými.

Hlavní předností nasazení programu PESMENU při tvorbě ovládní aplikace je mnohonásobné urychlení tvorby zdrojového textu, unifikace ovládní a v neposlední řadě i programátorské pohodlí. Programování ovládní aplikace se při použití nástroje PESMENU redukuje na návrh ovládacích prvků a jejich vlastností. Zbytek práce již zajistí návrhový systém automaticky.

2. APLIKACE A JEJÍ OVLÁDÁNÍ

Mezi základní úkony při aplikaci automatů systému PES patří tvorba a ladění uživatelského programu. Program se zapisuje ve formátu zdrojového textu obvykle pomocí editoru v prostředí PESPRO. V tomto prostředí probíhá též překlad, zatažení přeloženého kódu do automatů a jeho následné ladění. Nejčastěji je úloha napsání zdrojového textu rozdělena na dva v podstatě nezávislé problémy. Prvním z nich je ovládání aplikace pomocí klávesnice a zobrazování zvolených proměnných na displeji. Druhý problém, který se řeší, je vytvoření zdrojového textu programu pro ovládání vstupů a výstupů automatu (např. regulační algoritmy apod.).

Problém programu pro ovládání vstupů a výstupů je vždy do jisté míry specifický a závisí na konkrétním zadání. Není ho tedy možné jednoduchým způsobem zobecnit. Problém ovládání klávesnice a displeje je však možné (při dodržení zvolených pravidel) rozebrat a zobecnit. Tvorba zdrojového textu tak může být v tomto případě automatizována.

Princip ovládání aplikace

Jak již bylo zmíněno, můžeme uživatelský program rozdělit na část regulačních (výkonných) algoritmů a část sdružující ovládací a zobrazovací algoritmy. Pro ovládací a zobrazovací algoritmy zavedeme společný název **ovládací panel**. Pro část regulační a výkonnou zavedeme název **aplikace**. Na oba zavedené názvy se budeme odkazovat v průběhu textu. Ovládání aplikace můžeme z hlediska toku dat rozdělit na:

- data a povely směřující od ovládacího panelu do aplikace,
- data a povely směřující od aplikace do ovládacího panelu (nejčastěji na displej automatu).

Použijeme-li výše uvedené rozdělení zdrojového textu uživatelského programu, musíme vždy řešit programové rozhraní, realizující oba zmíněné směry toku dat. Pouze v případě definice tohoto rozhraní můžeme obě části zdrojového textu uživatelského programu řešit odděleně. K tomuto oddělenému řešení vždy vede použití programovacího nástroje PESMENU.

Princip ovládání aplikace je tedy založen na výměně dat pomocí programového rozhraní. Aby byla výměna dat vždy korektní a nedocházelo k jejich ztrátě, je nutné vždy využívat ovládací služby programového rozhraní v obou směrech toku dat. Ve směru od panelu do aplikace je korektní výměna dat zajištěna nástrojem PESMENU. V opačném směru tj. od aplikace do ovládacího panelu musí zajistit korektní výměnu dat uživatel.

Definice programového rozhraní

Použité programové rozhraní pro přenos dat mezi aplikací a ovládacím panelem je událostního typu. Definujeme tedy událost jako dvojici hodnot. Prvky této dvojice nazýváme:

- Zpráva (významová část)
- Data (parametrická část)

Při praktické realizaci definice události používáme pro každou z obou částí po jedné proměnné typu word (D proměnné). Proměnnou pro část „Zpráva“ označíme G_Message. Proměnnou pro část „Data“ označíme G_MessData. Vzhledem k tomu, že obě použité proměnné mohou zobrazovat čísla v rozsahu od 0 do 65535, je zřejmé, že můžeme od sebe odlišit max. 65536 zpráv a v rámci těchto zpráv můžeme přenášet hodnoty v rozsahu od 0 do 65535.

Rozsah proměnné pro přenášení G_Message zpráv dále rozdělíme na dvě poloviny, tj. od 1 do 32767 a od 32769 do 65535 a odlišíme tak směr toku dat. Zprávy s číslem od 1 do 32767 použijeme pro přenos dat od panelu do aplikace. Zprávy s číslem od 32769 do 65535 použijeme pro přenos dat ve směru opačném tj. od aplikace k panelu. Zpráva s číslem 0 je vyhrazena pro zprávu typu „není žádná zpráva“. Zpráva s číslem 32768 je vyhrazena pro obecnou zprávu typu „Překreslí“. Uvedme toto rozdělení ještě přehledně v tabulce:

Číslo zpráv	Směr toku dat
0	libovolný, na zprávu se nereaguje
1-32767	zpráva pro aplikaci, reaguje aplikace
32768	zpráva „Překreslí“ pro panel
32768-65535	zpráva pro panel, reaguje panel

Princip zpracování zprávy

Jak již bylo uvedeno, má vyslaná zpráva přiřazeno číslo. Toto číslo zprávy je unikátní pro danou zprávu a vztahuje se vždy pouze k jediné události (např. změna hodnoty, změna nastavení proměnné, změna stavu regulátoru apod.). Zpráva je specifikována vždy pouze pro jeden směr toku dat. Požadujeme-li přenos např. hodnoty zvolené proměnné v obou směrech používá se zvyklost, že se číslo zprávy pro panel liší od čísla zprávy pro aplikaci o hodnotu 32768.

Pro předání dat pomocí zprávy je nutné realizovat formátování zprávy v místě odkud je vysílána a zpracování zprávy tam, kde je přijímána. Vysílání zprávy spočívá v naplnění proměnné G_Message číslem zprávy a proměnné G_MessData parametrem, tj. hodnotou, pokud tuto hodnotu zpráva vyžaduje. Zpracování zprávy se provede, tak že po detekci zprávy známého čísla použijeme předaná data v proměnné G_MessData (pokud je zpráva předpokládá) a do proměnné G_Message запиšeme hodnotu 0, tj. zprávu typu „není žádná zpráva“. Uvedme příklad zápisu zpracování zprávy v jazyce SIMPLE:

```
TEPLOTA      #      D0
G_MESSAGE    #      D1
G_MESDATA    #      D2
32769        #      ZASTAV
.
; ZDE NÁSLEDUJE ZDROJOVÝ TEXT APLIKACE
.
IF (TEPLOTA > 25) THEN G_MESSAGE = ZASTAV
.
; ZDE JE PROGRAMOVÝ ŘÁDEK Z ČÁSTI PANELU,
; KTERÝM SE ZPRACUJE ZPRÁVA ZASTAV
.
IF (G_MESSAGE = ZASTAV) THEN 'YO : G_MESSAGE = 0
```

V uvedeném příkladu se předává zpráva z aplikace do panelu. Zpráva je nejprve detekována v podmínkové části podmíněného příkazu. Poté je zpracována tak, že nastaví logickou nulu na výstupu Y0 a poté je vynulována proměnná G_Message na znamení toho, že zpráva byla zpracována, a také proto, aby nebyla zpracována znovu v následujícím průchodu programovou smyčkou.

!Pozor! V příkladu je při zpracování zprávy použito ovládní výstupu Y0 pouze z důvodu, že smysluplné zpracování zprávy musí obsahovat nějakou akci. Ve skutečnosti se v části panelu žádný výstup či periferní obvod automatu neovládá. Pracuje se v něm jenom s hodnotami v paměti a s adresami těchto hodnot.

Na tomto místě je též nutné uvést zmínku o vyhrazené zprávě „Překreslí“ (zpráva č. 32768). Tato zpráva se vysílá z aplikace do ovládacího panelu. Ovládací panel na ni reaguje překreslením displeje. Nejvhodnější případy použití této zprávy najdete v dalších odstavcích textu. Zde by detailní popis zprávy vyzněl nesrozumitelně.

3. OVLÁDACÍ PRVKY APLIKACE V PESMENU

Základní systém ovládacích a zobrazovacích prvků, které předpokládá programovací nástroj PESMENU, obsahuje prvky těchto typů:

Prvky ovládací:

- Menu
- Menu s nadpisem
- Editor výčtu
- Editor hodnoty
- Příkaz

Prvky zobrazovací:

- Odkaz na výčet
- Odkaz na hodnotu
- Odkaz na uživatelský typ
- Obrazovka

Z těchto uvedených typů prvků je možné poskládat ovládací panel aplikace a vygenerovat zdrojový text panelu. Tento zdrojový text pak doplníme zdrojovým textem aplikace a s pomocí pravidel pro předávání dat (viz. Kap. 2) propojíme obě části do výsledného uživatelského programu.

3.1. Zobrazování na displeji a jeho princip

Všechny typy uvedených ovládacích prvků vyjma typu Příkaz využívají displej automatu k zobrazení stavu v němž se nacházejí, hodnot a dalších informací. Vzhledem k tomu, že zobrazovací proces trvá poměrně dlouho a příliš časté překreslování displeje vede k blikání vypisovaných textů, je pro zobrazování použit princip, který můžeme zkráceně označit jako „Piš když je potřeba“. Tento princip znamená, že překreslení displeje se vyvolá pouze tehdy, když nastane nějaká změna ve vykreslovaném textu na displeji.

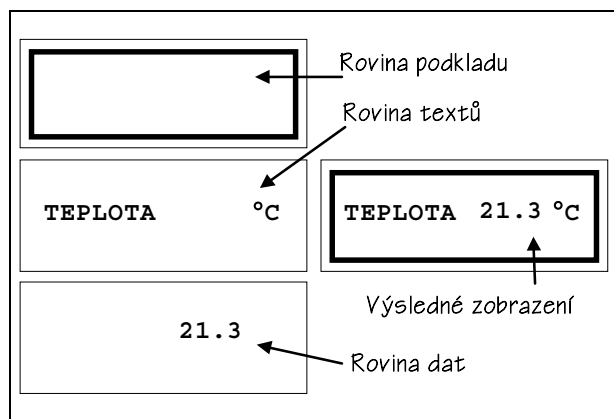
Vypisování textu na displej se formátuje ve třech rovinách (úrovních). Jednotlivé roviny označujeme:

- Rovina podkladu**
- Rovina textů**
- Rovina dat**

Nejprve se vykresluje na displej rovina podkladu, tj. sada textů, jejichž formát se nemění. Po té se vykreslí rovina textů. Nakonec se kreslí rovina dat. Jednotlivé roviny se tedy vykreslují přes sebe, tj. rovina dat přepisuje obě textové roviny, vypsané nejdříve.

Z uvedeného textu není na první pohled patrné, z jakého důvodu rozlišujeme rovinu podkladu a rovinu textů. Obě roviny vypisují konstantní texty a zdánlivě neexistuje rozumný důvod, proč je rozlišovat. Rozdíl tu však přeci je. Ne sice ve výsledném zobrazení, ale v oblasti návrhu textů v prostředí PESMENU. Zatímco rovinu podkladu editujeme pomocí myši, tj. kreslíme vlastně po obrazovce s pomocí zvolené znakové sady, rovinu textů editujeme textovým editorem. Každý vložený text do roviny textů, pak můžeme libovolně přesouvat po displeji a volit tak jeho umístění. V konečném výsledku se obě tyto roviny sloučí podle výše popsaného pravidla o přepisování. Výsledný zdrojový text generovaný pro vykreslování roviny podkladu a roviny textů pak vykresluje obě roviny současně.

Na obrázku Obr. 3-1 jsou uvedeny příklady všech tří rovin včetně jejich sloučení a výsledného výpisu na displej automatu. V rovině dat se v našem příkladu vypisuje hodnota proměnné, obsahující údaj o teplotě v místnosti.



Obr. 3-1 Zobrazovací roviny výpisu na displej

Rovina podkladu a rovina textů

Rovina podkladu a rovina textů může obsahovat pouze neměnné, tj. konstantní znakové výpisy. V případě roviny podkladu jsou znaky umísťovány myší a je možné je pouze smazat. Každý z umístěných znaků je chápán v programu PESMENU samostatně. V rovině textů jsou znaky sdruženy do textových řádků či úseků. Takto definované úseky je možné libovolně přesouvat a umísťovat po ploše displeje. Pokud jsme s jejich umístěním spokojeni, pak je jejich umístění chápáno jako konstantní.

Rovina dat

Rovina dat je určena pro rozmístění výpisů proměnných, jejichž zobrazení na displeji požadujeme. Při definici výpisu proměnné volíme jednak umístění výpisu a jednak vyhrazujeme počet znaků, do nichž bude výpis umístěn. Vyhrazené místo pro výpis je určující v případě zobrazení proměnné typu výčet (převod aktuální hodnoty proměnné na předdefinovaný text). Pro zobrazování proměnné typu hodnota (číselné zobrazení) se však výsledný počet znaků použitých pro výpis řídí zadaným číselným formátem pro výstup na displej.

3.2. Ovládací prvek MENU a MENU S NADPISEM

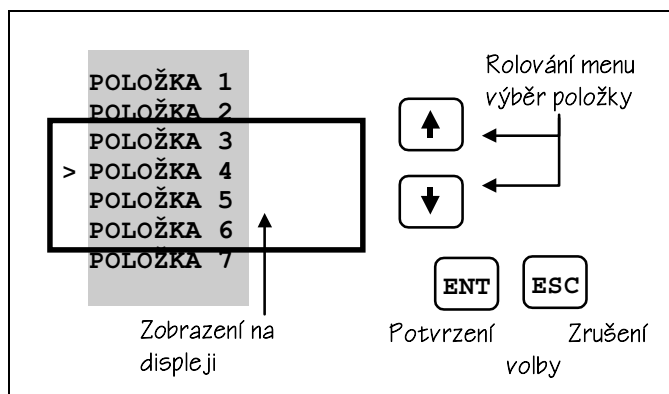
Ovládací prvek MENU je základním stavebním kamenem ovládacího panelu aplikace. Menu se skládá z jednotlivých položek menu, které jsou označeny textem a u nichž se předpokládá při jejich volbě vyvolání specifické akce (např. zapnutí motoru). Akcí, kterou může položka menu vyvolat, odpovídá výše definovaným ovládacím prvkům. Je tedy zřejmé, že akcí může být:

- Vyvolání menu**
- Vyvolání menu s nadpisem**
- Vyvolání editoru výčtu**
- Vyvolání editoru hodnoty**
- Vyvolání prvku obrazovky**
- Vyvolání příkazu**

Ostatní prvky tj. odkazy všech typů vyvolány být nemohou, neboť se jedná o zobrazovací prvky používané jako podpůrné.

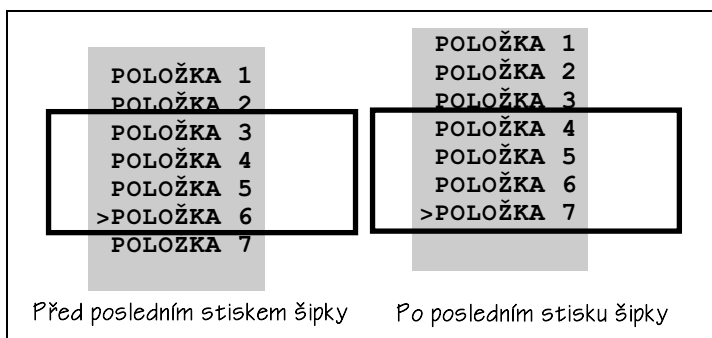
Uspořádání a ovládání menu

Menu sestává obecně z libovolného počtu položek. Vzhledem k tomu, že displeje automatů řady K jsou čtyřřádkové a řady MPC300 dokonce dvouřádkové, je vhodné volit počet položek menu právě s ohledem na počet řádků displeje. Při velkém počtu položek menu se totiž stává procházení těmito položkami nepřehledné.



Obr. 3-2 Ovládání menu a výběr položky

Ovládání prvku menu je realizováno pomocí kláves se šipkami a kláves Enter a Esc. Šipkami nahoru a dolů posouváme kurzor vybrané položky menu, případně jimi rolujeme. Rolování položek je zajištěno automaticky v krajních polohách kurzoru. Klávesou Enter potvrzujeme volbu. Klávesou Esc opouštíme zvolené menu. Na obrázku Obr. 3-2 je situace znázorněna graficky.



Obr. 3-3 Rolování položek menu

Z obrázku Obr. 3-2 je zřejmé, že menu obsahuje celkově 7 položek. Všechny tyto položky není možné současně zobrazit na displeji a proto, je potřeba zajistit rolování

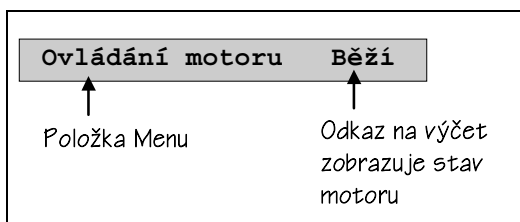
menu. To probíhá automaticky tak, že posuneme-li šipkou dolů kurzor na poslední zobrazovanou položku a provedeme další stisk šipky. Posunou se položky menu, nikoliv umístění kurzoru. Výsledné zobrazení je uvedeno na Obr. 3-3.

Opačná situace nastane v případě posunu kurzoru na první zobrazovanou položku menu. Po stisku šipky nahoru budou položky menu rolovat směrem dolů. V obou případech bude rolování zastaveno na krajních položkách menu.

Položka menu a její parametry

Každá položka menu je určena svým zobrazením, tj. podkladem a textem na displeji automatu a dále pak akcí, kterou její volba vyvolává. V zobrazení je možné využít i rovinu dat. Do této roviny umístíme odkazy. Odkazy mohou být

všechny typů, tj. odkaz na hodnotu, výčet nebo uživatelský typ. Situaci znázorňuje Obr. 3-4. Na tomto obrázku je uvedena jedna položka z menu. Text položky „Ovládání motoru“ označuje akci, kterou položka vyvolá. Jedná se o nastavení stavu motoru, tj. např. zastavení, rozběh apod..



Obr. 3-4 Příklad definice položky menu

vení, rozběh apod..

V zobrazovací rovině dat je do položky vložen odkaz na výčet. V uvedeném příkladu tento odkaz zobrazuje aktuální stav motoru. V případě, že chceme tento stav změnit, nastavíme kurzor menu na tuto položku a stiskneme klávesu Enter. Tím se vyvolá akce nastavení stavu motoru. V případě, že stav motoru změníme a vrátíme se zpět do menu, bude odkaz u uvedené položky zobrazovat nový stav motoru.

Ovládací prvek MENU S NADPISEM

Ovládací prvek MENU S NADPISEM je odvozen od prvku MENU. Odlišnost spočívá v tom, že první řádek displeje je použit na zobrazení konstantního textu, tj. jakési hlavičky MENU. Tento řádek není posouván při rolování položek MENU a zůstává vždy zobrazen. Výhodou tohoto uspořádání je snadnější orientace při listování ve stromové struktuře menu. Nevýhodou je pak snížení počtu řádků pro zobrazování položek menu. Tato nevýhoda se uplatní zvláště u automatů řady MPC300, které mají displej pouze dvouřádkový.

3.3. Ovládací prvek EDITOR VÝČTU a ODKAZ NA VÝČET

Ovládací prvek EDITOR VÝČTU a zobrazovací prvek ODKAZ NA VÝČET slouží k editaci a zobrazování proměnné typu výčet. Jedná se o proměnnou, u níž se

předpokládá, že nabývá několika hodnot. Tyto hodnoty obvykle vyjadřují nějaký stav zařízení.

Nejjednodušší proměnnou typu výčet si můžeme představit ve formě spínače. Spínač má dva stavy, které můžeme slovně vyjádřit slovy „sepnut“ a „rozpojen“. Číselně můžeme hodnoty vyjádřit např. hodnotou 0 pro stav „rozpojen“ a hodnotou 1 pro stav „sepnut“. Jiných stavů spínač nabývat nemůže.

Větší počet stavů můžeme definovat např. pro přepínač, když slovně popíšeme jeho jednotlivé polohy. Příkladem může být přepínač vlnových rozsahů u rozhlasového přijímače. Zde bývají polohy označeny SV, DV, KV, VKV. Takto definovaný přepínač může nabývat čtyř stavů, které můžeme vyjádřit přirozenou řadou čísel od 0 do 3.

V případě, že využijeme analogii přepínače pro zobrazování na displej, můžeme místo číselných hodnot u zvolené proměnné zobrazovat texty odpovídající té či oné hodnotě proměnné. V programovacím nástroji PESMENU se pro tento typ zobrazování předpokládá přiřazení maximálně deseti textů. Z tohoto předpokladu tedy vychází číselný rozsah proměnné typu výčet. Hodnota této proměnné se tedy může pohybovat v rozsahu od 0 do 9. V případě, že chceme aktuální hodnotu proměnné zobrazit, můžeme na proměnnou vytvořit tzv. odkaz. V případě, že požadujeme možnost editace této proměnné, musíme použít editor výčtu.

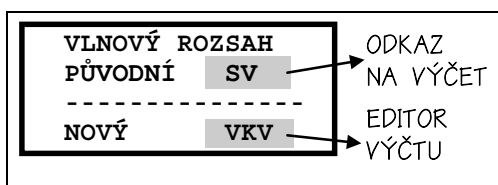
Uspořádání a ovládání editoru výčtu

Pro vysvětlení uspořádání a ovládání editoru výčtu použijeme již uvedený příklad s přepínačem vlnových rozsahů. Zmíněný přepínač chceme ovládat z klávesnice automatu. Použijeme tedy jako akci položky menu vyvolání editoru výčtu. Můžeme ji nazvat např. „Vlnový rozsah“ a přiřadíme ji akci vyvolání editoru výčtu.

Editor výčtu využívá pro editaci kláves šipka nahoru a dolů pro změnu hodnoty z předchozí na následující a zpět. Dále pak klávesu Enter pro potvrzení nastavené hodnoty s následným opuštěním editoru a klávesu Esc pro opuštění editoru bez změny hodnoty proměnné. Pro uchování hodnoty proměnné se používá proměnná typu word. Při změně hodnoty se modifikuje též proměnná G_Message a proměnná G_MessData z programového rozhraní, aby bylo možné v aplikaci na tuto změnu proměnné reagovat.

Pro použití editoru výčtu je nutné provést přiřazení textů hodnotě proměnné a navrhnout vzhled obrazovky editoru. Obrazovka editoru, tj. formát výpisu na displej je, jak už bylo popsáno, složena z rovin podkladu a textů a z roviny dat. Vhodné uspořádání editoru výčtu ukazuje Obr. 3-5. Vidíme zde uspořádání na obrazovce editoru, které se skládá z textů pro název proměnné, označení původní hodnoty a označení nové hodnoty. Všechny tyto texty jsou vloženy v rovině podkladu a v rovině textů.

V rovině dat je vloženo pole editoru výčtu, které vkládá program PESMENU automaticky a nelze ho ze zobrazení odstranit. Odkaz na výčet je vložen již ručně a



v zobrazení již být nemusí. Použití odkazu je však v naprosté většině případů opodstatněné, neboť je vhodné, má-li obsluha při editaci hodnoty k dispozici její aktuální nastavení.

Obr. 3-5 Typický formát editoru výčtu

Odkazů na různé proměnné je možné do zobrazení vložit větší množství podle toho, je-li to pro vhodnou volbu hodnoty pomocí editoru potřeba. Všechny prvky z datové roviny můžeme libovolně umísťovat po ploše displeje, zvětšovat nebo zmenšovat. Mazání je dovoleno u všech, vyjma pole editoru výčtu.

Zobrazovací prvek ODKAZ NA VÝČET

Jak bylo ukázáno v příkladu uvedeném v předchozím odstavci, slouží zobrazovací prvek odkazu na výčet k zobrazování aktuálně nastavené hodnoty dané proměnné. S proměnnou je svázán jménem a toto jméno je považováno za jedinečné. Odkaz tedy slouží k zobrazení zvolené proměnné, jejíž jméno nese v libovolném zobrazení ovládacího panelu.

Odkaz může být součástí položky displeje, zobrazovacího prvku obrazovka, editoru výčtu nebo hodnoty atd.. Při návrhu zobrazení editoru výčtu je, jak již bylo uvedeno, vhodné doplnit editor o odkaz na editovaný výčet a zobrazit tak původní nastavení editované proměnné.

Odkaz na výčet je možné použít i samostatně, tj. bez vazby na editor výčtu. V takovém případě slouží odkaz pouze pro zobrazování hodnoty proměnné, obecně vzato stavu nějakého nastavení či procesu v aplikaci. Je-li výčet definován takto samostatně, musíme pro něj vytvořit texty, označující jednotlivé hodnoty přiřazené proměnné. V případě, že je odkaz svázán s editorem, jsou dosazeny texty stavů, definované v editoru výčtů.

3.4. Ovládací prvek EDITOR HODNOTY a ODKAZ NA HODNOTU

Ovládací prvek EDITOR HODNOTY a zobrazovací prvek ODKAZ NA HODNOTU slouží k editaci a zobrazování proměnné typu hodnota. Jedná se o proměnnou, u níž se předpokládá, že bude editována a zobrazována v jednom z 16-ti základních typů zobrazení, které automaty umožňují. Proměnná obsahující editovanou a zobrazovanou hodnotu je typu word a zvolený formát zobrazení určuje její interpretaci, tj. zda se bude zobrazovat znaménkově či neznaménkově, a který dekadický řád bude určovat pozici pro výpis desetinné tečky.

Uspořádání a ovládání editoru hodnoty

Příkladem pro použití editoru hodnoty může být úloha nastavení teploty v místnosti. Pro teplotu v místnosti zvolíme zobrazovací formát na jedno desetinné místo (formát č. 5 viz katalog Programovatelné logické automaty MPC300 a PES-K). Budeme tedy editovat a zobrazovat teplotu s přesností v desetinách stupně Celsia. Dále omezíme rozsah pro nastavení hodnoty minimální a maximální povolenou hodnotou např. od 2.0 do 28.0 tj. pro proměnnou typu word ve zvoleném formátu zobrazení omezíme hodnotu rozsahem od 20 do 280. Editor hodnoty pro nastavení teploty v pokoji vložíme do menu např. jako akci položky „TEPLOTA“. Vhodné uspořádání editoru pro nastavení teploty ukazuje Obr. 3-6.



Obr. 3-6 Typické uspořádání editoru hodnoty

Editor má v obou případech vestavěnou kontrolu rozsahu editované proměnné. V případě, že zadáme hodnotu mimo rozsah nebo ve špatném formátu, je vyvoláno na displeji chybové hlášení a nová hodnota není přijata. Chybové hlášení můžeme smazat a vrátit se tak k editaci stiskem některé z číselných a editačních kláves nebo stiskem klávesy šipka vlevo či vpravo. Zobrazení je i v tomto případě doplněno odkazem na hodnotu pro zobrazení původně nastavené hodnoty v průběhu editace.

Editor má v obou případech vestavěnou kontrolu rozsahu editované proměnné. V případě, že zadáme hodnotu mimo rozsah nebo ve špatném formátu, je vyvoláno na displeji chybové hlášení a nová hodnota není přijata. Chybové hlášení můžeme smazat a vrátit se tak k editaci stiskem některé z číselných a editačních kláves nebo stiskem klávesy šipka vlevo či vpravo. Zobrazení je i v tomto případě doplněno odkazem na hodnotu pro zobrazení původně nastavené hodnoty v průběhu editace.

Při úspěšném nastavení hodnoty a stisku klávesy Enter je proměnná přepsána novou hodnotou a současně jsou modifikovány proměnné G_Message a G_MessData, aby bylo možné v aplikaci na změnu hodnoty proměnné reagovat. Dále je nutné upozornit na to, že výsledný počet znaků displeje, který editor a odkaz použije pro zobrazení hodnoty se v tomto případě vždy řídí zvoleným formátem zobrazení a nikoliv velikostí příslušného pole v rovině dat.

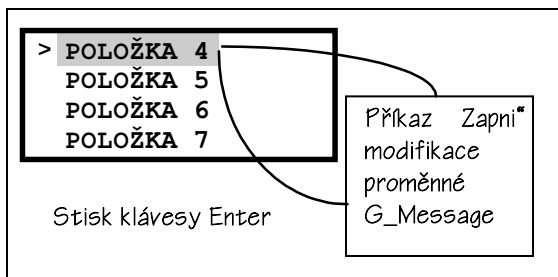
Zobrazovací prvek ODKAZ NA HODNOTU

V popisovaném příkladu z předchozího odstavce je zobrazovací prvek odkazu na hodnotu použit k zobrazení aktuální hodnoty proměnné v daném formátu. Odkaz na hodnotu je možné použít i v položce menu či v zobrazovacím prvku obrazovka.

S příslušným editorem je možné odkaz svázat přes jméno proměnné, které je považováno za jedinečné. Z toho plyne, že všechny použité odkazy téhož jména zobrazují tutéž hodnotu v tomtéž formátu. V případě, že k odkazu daného jména neexistuje editor, je odkaz chápán jako samostatný a hodnotu, kterou zobrazuje tedy není možné editovat.

3.5. Ovládací prvek PŘÍKAZ

Ovládací prvek příkaz je nejjednodušší ovládací prvek, s nímž programovací nástroj PESMENU pracuje.



Oproti ostatním ovládacím prvkům má dvě zásadní odlišnosti. Neexistuje k němu zobrazení na displeji, nemá přiřazenou proměnnou a bezprostředně po vyslání zprávy přes programové rozhraní vrací řízení menu obsahující položku, která ho aktivuje. Z menu se vyvolává stiskem klávesy Enter. Po aktivaci ovládací prvek příkaz naplní hodnotu

Obr. 3-7 Princip použití prvku „Příkaz“

proměnné G_Message číslem příkazu a vrátí řízení zpět do menu, z něhož byl aktivován.

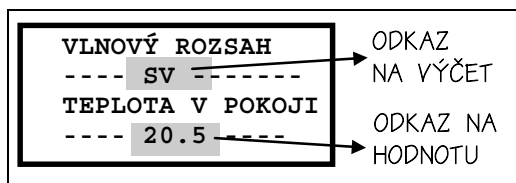
Situaci znázorňuje obrázek Obr. 3-7. Položka menu označená v obrázku „Položka 4“ má definovanou akci volání ovládacího prvku příkaz. Nastavíme-li kurzor menu na tuto položku a stiskneme-li klávesu Enter, bude vyvolán právě ovládací prvek příkaz. Ovládací prvek příkaz запиše do proměnné G_Message své číslo a vrátí řízení zpět do menu. Tímto způsobem je k aplikaci vyslána informace o tom, že byl vyvolán zmíněný příkaz a aplikace tento příkaz zpracuje. Pokud nedojde ke zpracování příkazu, je tento příkaz v dalším kole volání ovládacího panelu smazán a zapomenut.

Ovládací prvek příkaz je v systému PESMENU identifikován jménem, které musí být jedinečné. Pokud tak není, je příkaz generován z několika míst menu. Ovládací prvek příkaz nemá přidělené žádné zobrazení na displeji a tudíž mimo názvu příkazu se žádné další parametry nenastavují.

3.6. Zobrazovací prvek OBRAZOVKA

Zobrazovací prvek Obrazovka slouží k zobrazování většího počtu hodnot. Umožňuje tak konstrukci přehledového panelu pro zobrazování skupin parametrů a nastavení aplikace. Prvek Obrazovka není určen k nastavování a editaci proměnných a tudíž má velmi jednoduché ovládání. Stiskem klávesy Enter či Esc předá řízení do menu, z něhož byl vyvolán stisky ostatní klávesy nejsou použity a jejich stisk není vyhodnocován. Zobrazení prvku obrazovka je složeno z rovin podkladu a textů a z roviny dat.

Do roviny dat umístíme pole odkazů na hodnotu, výčet nebo pole odkazu na uživatelský typ. Počet umístěných odkazů není limitován, nicméně je vhodné navrhovat zobrazení tak, aby bylo přehledné. V případě, že používáme odkazy na výčet nebo hodnotu,



Obr. 3-8 Příklad prvku "Obrazovka"

ke kterým existují editory, identifikujeme je pouze jménem. Zbytek parametrů doplní PESMENU v průběhu generování výsledného zdrojového textu podle příslušného editoru. Pokud k odkazům neexistují, editory je nutné doplnit i zbývající potřebné parametry. Pro příklad (viz. Obr. 3-8) zobrazovacího prvku Obrazovka jsou použity odkazy, vysvětlené v příkladech editoru výčtu a editoru hodnoty.

3.7. Odkaz na UŽIVATELSKÝ TYP

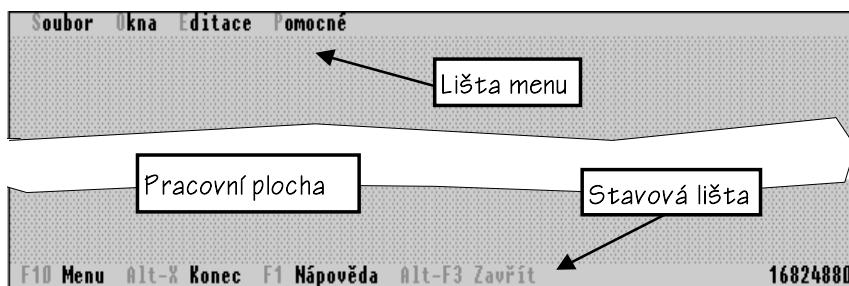
Až dosud jsme se zmiňovali pouze o prvcích a typech, které jsou v nástroji PESMENU jednoznačně definovány. Jsou však situace, kdy tyto předdefinované ovládací a zobrazovací prvky nepokryjí všechny potřeby v ovládní aplikace. Z tohoto důvodu pracuje programovací nástroj PESMENU s virtuálním prvkem uživatelského typu.

Tento typ je v PESMENU chápán jako odkaz, protože není dopředu znám editační model a formát zobrazení. Proto PESMENU v tomto případě pracuje pouze s obecně platnými parametry prvku, tj. se jménem, umístěním a velikostí pole zobrazení na displeji. Pro uživatelský typ vygenerována pouze programová kostra.

4. POPIS PROGRAMU PESMENU

Programovací nástroj PESMENU je řešen jako doplňkový program pro prostředí PESPRO. Nástroj PESMENU je možné spouštět samostatně a nebo z položky menu „Nástroje“ z prostředí PESPRO. Obvyklý postup při používání programu PESMENU je odvozen z programování běžného uživatelského programu pro automaty systému PES. Při programování aplikace obvykle postupujeme tak, že analyzujeme zadání problému, uvážíme požadavky na ovládání automatu v daném aplikaci, vybereme řídicí a regulační algoritmy a zadání zrealizujeme. Při programování aplikace zmíněného typu můžeme problém rozdělit na problém ovládání a na problém vlastních regulačních a řídicích algoritmů. Zatímco problém řídicích a regulačních algoritmů bývá většinou specifický, tak ovládání aplikace tj. ovládací panel vykazuje většinou významné podobnosti a to i v aplikacích různých typů a z různých oborů. Podobností v ovládání aplikací je možné s úspěchem využít a usnadnit si tak práci s tvorbou části zdrojového textu, která řeší právě ovládací panel aplikace. Pro usnadnění programování ovládacího panelu je určen programovací nástroj PESMENU a jeho nasazení je možné v tomto úseku programátorské práce jen doporučit.

Práci s programovacím nástrojem je možné rozdělit do dvou kroků. Prvním krokem je návrh ovládání. Jedná se o návrh struktury ovládání, návrh zobrazení údajů a návrh editorů proměnných. Tento krok je nutné udělat ručně za podpory nástrojů implementovaných v PESMENU. Druhý krok je již plně automatizován a spočívá ve spuštění generátoru zdrojového textu, který je součástí programu PESMENU. Výsledný zdrojový text se poté načte do textového editoru prostředí PESPRO, přeloží a zatáhne do automatu. V této fázi sice ještě nejsou implementovány regulační a řídicí algoritmy, nicméně je možné zkusit komfort a vlastnosti ovládání aplikace. V případě, že je potřeba na ovládání něco upravit, spustíme znovu nástroj PESMENU, provedeme úpravy a znovu vygenerujeme zdrojový text. Ten znovu přeložíme a vyzkoušíme. Jsme-li spokojeni, zbývá dodat regulační a řídicí algoritmy a navázat je na vyzkoušený zdrojový text aplikace.



Obr. 4-1 Základní obrazovka PESMENU

4.1. Základní obrazovka a příkazy menu

Na Obr. 4-1 je zobrazena základní obrazovka programovacího nástroje PESMENU. Obrazovka se skládá z pracovní plochy, lišty menu a stavové lišty doplněné o pole „horkých“ kláves. Lišta *Menu* je složena z položek *Soubor*, *Okna*, *Editace* a *Pomocné*. Položka *Soubor* je určena pro práci se soubory ve formátu programovacího nástroje PESMENU. Položka *Okna* sdružuje základní příkazy pro umístění a ovládání oken editoru souborů. Položka *editace* obsahuje příkazy pro vkládání, mazání a editaci jednotlivých položek editovaného souboru. Jsou zde i příkazy pro definici znakové sady, použité v souboru a příkaz pro generování výstupního zdrojového textu. Položka *Pomocné* zahrnuje příkazy, týkající se nastavení prostředí. Stavová lišta obsahuje pole horkých kláves pro vstup do lišty menu (F10), ukončení programu (Alt-X), uzavření okna editoru (Alt-F3) a vyvolání nápovědy (F1). Dále je na stavové liště pole s informací o velikosti dostupné paměti, tj. paměti, kterou má PESMENU k dispozici.

Položka menu SOUBOR

Položka menu *SOUBOR* (Obr. 4-2) obsahuje příkazy pro práci se soubory. Tyto příkazy jsou doplněny o příkazy ukončení práce v programu (Konec - Alt-X), změnu pracovního adresáře (Adresář), vyvolání obrazovky systému DOS (DOS). Příkazy pro práci se soubory mají tento význam:

Nový	F4
Otevři	F3
Ulož	F2
Ulož se jménem...	Shift-F2
Adresář	
DOS	
Konec	Alt-X

Obr. 4-2 Položka menu SOUBOR

- Nový** příkaz založí nový soubor se standardním nastavením
- Otevři (F3)** příkaz otevře již existující soubor z disku podle volby z dialogového okna pro otevření souboru
- Ulož (F2)** příkaz provede uložení souboru z aktivního okna editoru
- Ulož se jménem (Shift-F2)** příkaz uloží obsah aktivního okna editoru pod zadaným jménem

Položka menu OKNA

Položka menu *OKNA* (Obr. 4-3) sdružuje příkazy pro práci s rozmístěním a uspořádáním oken editorů menu. Jednotlivé příkazy z nabídky *OKNA* mají následující význam:

- Další** (F6) příkaz vybere z pracovní plochy okno editoru menu, které následuje v řadě za právě aktivním oknem a umožní tak editaci obsahu okna.
- Zvětšení** (F5) je-li aktivní okno menší než rozměr pracovní plochy, pak ho příkaz zvětší na rozměr této plochy, je-li naopak okno roztazeno po celé ploše, pak ho příkaz zmenší na předchozí velikost.
- Uzavření** (Alt-F3) příkazem odstraníme okno editoru menu z pracovní plochy.
- Rozměr** (Ctrl-F5) příkazem se mění rozměr okna editoru menu pomocí klávesnice. Po vyvolání příkazu měníme umístění okna po pracovní ploše klávesami šipek, rozměr okna upravujeme stiskem klávesy Shift společně s klávesami šipek.
- Vedle sebe** v případě většího počtu oken editorů menu na pracovní ploše rozmístí příkaz tyto okna vedle sebe za současného upravení jejich rozměrů tak, aby se vedle sebe okna vešla.
- Za Sebou** příkaz uspořádá okna editorů do podoby lístků kartotéky.



Obr. 4-3 Položka menu OKNA

Položka menu EDITACE

Položka menu *Editace* (Obr. 4-4) sdružuje příkazy pro editaci v editoru menu. Tyto příkazy zastupují hlavní editační funkce, dají se vyvolat z klávesnice. Základní příkazy pro editor menu jsou doplněny příkazy pro úpravu znakové sady, pro způsob zobrazení a příkazem pro spuštění generátoru zdrojového textu. Jednotlivé příkazy mají následující význam:

- Položka** (Ins) příkaz vyvolá doplňkové menu s nabídkou prvků menu, které je možné vložit na místo kurzoru v editoru menu. Příkaz se vyvolá stiskem klávesy Ins z editoru menu.



Obr. 4-4 Položka menu EDITACE

- ❑ **Objekt** (Shift-Ins) příkaz je obdobou příkazu položka a pokud to umístění kurzoru umožňuje vyvolá pomocné menu s nabídkou prvků (objektů), které je možné vložit na danou pozici kurzoru. Příkaz se vyvolá stiskem kláves Shift-Ins.
- ❑ **Smaž** (Del) příkaz smaže položku z editoru menu na níž je umístěn kurzor. Příkaz se vyvolá stiskem klávesy Del.
- ❑ **Nastavení** (Ctrl-Enter) příkaz vyvolá dialogové okno pro nastavení parametrů položky či prvku na němž se nachází kurzor. Příkaz se vyvolá stiskem kláves Ctrl-Enter.
- ❑ **Popis** (Alt-D) příkaz přepíná zobrazovací režim editoru menu z přehledu položek menu na detailní popis položky, na níž je umístěn kurzor a naopak. Příkaz se vyvolá stiskem kombinace kláves Alt-D. V případě návratu k přehledovému zobrazení je možné použít také klávesu Esc.
- ❑ **Znaková sada** (Alt-C) příkaz vyvolá dialogové okno s editorem znakové sady pro úpravu tvaru uživatelsky nastavitelných znaků. Příkaz se vyvolá stiskem kombinace kláves Alt-C
- ❑ **Překlad** (Alt-F9) příkaz slouží ke spuštění generátoru zdrojového textu, tj. k převodu navrženého menu z formátu editoru menu na výstupní zdrojový text, zpracovatelný překladačem prostředí PESPRO. Příkaz se vyvolá stiskem kombinace kláves Alt-F9.

Položka menu POMOCNÉ

Položka *POMOCNÉ* obsahuje příkazy pro nastavení prostředí PESMENU a doplňkové příkazy pro zobrazení informačního okna a příkaz pro vyvolání indexu nápovědy. Jednotlivé příkazy položky *POMOCNÉ* mají tyto významy:

- ❑ **Prostředí** příkaz otevře dialogové okno se základními parametry nastavení prostředí a generátoru výstupního zdrojového



Obr. 4-5 Položka menu POMOCNÉ

textu.

- ❑ **Barvy** příkaz vyvolá dialogové okno pro nastavení barev zobrazení jednotlivých ovládacích a zobrazovacích prvků prostředí.
- ❑ **O Programu** příkazem se vyvolá informační okno, obsahující základní údaje o verzi produktu. Jsou zde uvedeny i některé doplňkové informace.
- ❑ **Nápověda Index** příkaz vyvolá základní soupis nápovědy.
(Ctrl-F1)

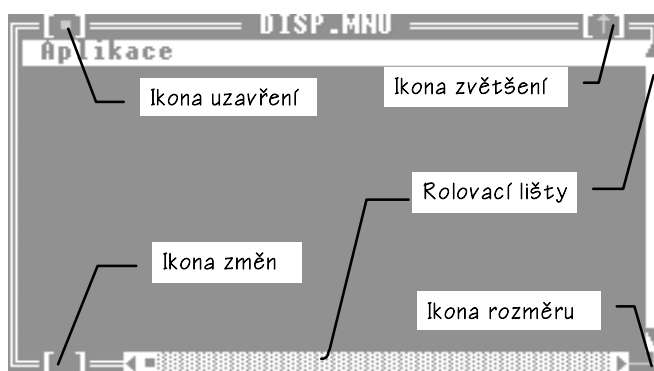
4.2. Editor menu

Editor menu tvoří základní prvek programovacího nástroje PESMENU. Je určen k zobrazení struktury návrhu menu a úpravám této struktury.

Ovládání editoru menu

Editor menu je tvořen základním oknem (

Obr. 4-6), vybaveným vodorovnou a svislou rolovací lištou, ikonou pro indikaci změn, ikonou zvětšování a zmenšování rozměrů okna, ikonou uzavření okna a ikonou pro



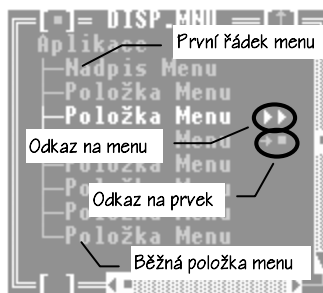
Obr. 4-6 Okno editoru menu

plynulou změnu velikosti okna. Všechny tyto ovládací prvky reagují na kurzor myši a stisk jejího levého tlačítka. V okně editoru menu je zobrazován kurzor, sloužící k procházení navrhovanou strukturou menu a k označování položky, určené k editaci.

K procházení strukturou položek editoru používáme řádkový kurzor, který ovládneme pomocí šipek. Základní zobrazení editoru zobrazuje zvolenou úroveň menu tak, že na prvním řádku je vypsána položka menu o úroveň vyšší. Za ní na dalších řádcích začínají položky menu zvolené úrovně. Pokud je zobrazována nejvyšší úroveň menu, tak je na prvním řádku vypsána položka Aplikace. Tato položka je základní, není jí možné smazat a obsahuje informace o následující struktuře menu. Povinná položka Aplikace je taktéž

součástí nového souboru založeného příkazem *NOVÝ* z položky *SOUBOR* na ovládací liště menu nástroje PESMENU. Tuto základní situaci zobrazuje Obr. 4-6.

Zobrazení struktury menu, která již obsahuje položky návrhu ukazuje. Jak je vidět z obrázku Obr. 4-7, základní zobrazení struktury menu je tvořeno položkami menu a nebo speciální položkou *Nadpis menu*, která však může být umístěna pouze na prvním řádku zobrazované struktury. Mírně tím první řádek, určený pro položky dané úrovně menu. Je-li struktura menu takto zadána, bude menu generováno jako ovládací prvek *MENU S NADPISEM* (viz. strana 10). Není-li položka *Nadpis menu* použita, pak bude generován ovládací prvek *MENU* (tj. menu bez nadpisu).



Obr. 4-7 Zobrazení struktury menu

Jednotlivé položky menu jsou zobrazovány buď přednastaveným názvem, jak je to uvedeno na obrázku a nebo přímo tak, jak budou zobrazovány na displeji automatu. Toto reálné zobrazení odpovídající zobrazení na displeji automatu je použito automaticky vždy, pokud je pro danou položku definováno.

Výpis položky menu může být doplněn o odkaz upozorňující na vnořenou strukturu, tj. na strukturu, kterou položka menu bude po zvolení volat. Pro tyto odkazy jsou zvoleny dva typy zobrazení a jsou použity podle toho, jaká struktura je položkou volána. V případě, že tuto navazující strukturu tvoří další menu, pak jsou pro odkazy použity znaky „>>“. Ve všech ostatních případech navazujících struktur jsou použity znaky „>■“. Pod ostatními navazujícími strukturami máme na mysli všechny ovládací prvky, které nejsou typu *MENU* nebo *MENU S NADPISEM*. Rozdělení ovládacích prvků na uvedené skupiny má ten význam, že jakýkoliv jiný prvek než je *MENU* nebo *MENU S NADPISEM* nemůže volat další navazující strukturu.

Procházení strukturou menu, vkládání a mazání položek

Pro procházení strukturou menu jsou vyhrazeny klávesy:

- Enter** vstup do následující či předchozí úrovně menu.

Vstup do následující úrovně menu je povolen pouze tehdy, má-li označená položka odkaz na následující strukturu. Návrat do předchozí úrovně menu je možný stiskem klávesy **Enter** na prvním řádku zobrazení (řádek je vyhrazen pro zobrazení položky menu o úroveň výš) pokud se nejedná o nejvyšší úroveň menu (první řádek má název *APLIKACE*).

- Šipky** \updownarrow pohyb kurzorem ve zvolené úrovni menu.
- Ins** vloží novou položku do struktury menu.

Pokud vkládáme první položku v nové úrovni menu, nabídne editor všechny typy dostupných ovládacích prvků. V případě, že zvolíme ovládací prvek editor výčtu, editor hodnoty, obrazovka nebo příkaz, vložení další položky již není možné. V případě, že jako první položku vložíme prvek menu, tj. položku menu nebo nadpis menu, můžeme dalším stiskem klávesy Ins přidat další položku menu. Položka typu nadpis menu může být vložena pouze jednou a to na první místo v aktuální struktuře menu. Editor menu všechny uvedené podmínky kontroluje, takže se jimi nemusíme zabývat. V tomto textu jsou uvedeny pouze pro vysvětlení vzájemných vazeb.

- ❑ **Del** klávesou Del smažeme libovolnou položku menu.

Klávesa Del slouží k mazání položek menu. Zde však musíme dávat pozor, neboť smazáním položky mažeme nejen tuto položku, ale současně všechny položky, které vyvolává, tj. které jsou ve všech nižších úrovních. Má-li tedy položka menu zobrazenou značku odkazu, vymažeme položku včetně všech struktur, které jsou schovány v tomto odkazu.

- ❑ **Ctrl-Enter** klávesou Ctrl-Enter vyvoláme editor zobrazení položky.

Editor zobrazení položky je realizován pomocí speciálního dialogového okna, které simuluje reálný vzhled displeje automatu. Zobrazení na tomto displeji se děje ve třech rovinách, jak bylo popsáno v odstavci „Zobrazování na displeji a jeho princip“ na straně 8. Stejný způsob tvorby zobrazení modeluje i tento editor zobrazení.

Zobrazovací režimy editoru menu

Editor menu umožňuje dva základní typy zobrazení. První typ byl ukázán na Obr. 4-7 Zobrazení struktury menu. Jedná se o stromové zobrazení aktuální úrovně menu. Druhým typem je zobrazení detailních informací o položce na níž stojí kurzor. Tento typ zobrazení má specifický tvar podle toho, kterou položku zobrazuje.

Na Obr. 4-8 je uveden příklad detailního zobrazení položky menu. Na prvním řádku je uvedeno jméno položky (buď implicitní a nebo zadané) společně tzv. unikátním číslem, podle něhož si programovací nástroj PESMENU třídí seznam struktur. Za oddělovací čarou následuje řádek s popisem akce, tj. s tím, co je stane při aktivaci této položky menu.



V uvedeném příkladě na obrázku bude akcí vyvolání dalšího menu neboť v popisu této akce je uvedeno „Nadpis menu“. Další

Obr. 4-8 Příklad detailního zobrazení

řádky za uvozujícím textem „Zobrazení“ jsou věnovány simulovanému textu zobrazení položky ve tvaru, v němž bude položka zobrazena na skutečném displeji. Zobrazení je doplněno po levé straně svorkou, aby byl patrný svislý rozměr skutečného displeje. Obdobně jsou řešena zobrazení dalších ovládacích prvků ze struktury menu.

Druhy ovládacích panelů podporované editorem menu

Editor menu podporuje tvorbu dvou odlišných druhů ovládacích panelů. První druh panelu je svázán s ovládáním aplikace. Vytváří systém menu, editorů hodnot, editorů výčtu, příkazů, odkazů a obrazovek. Nazýváme ho úplným modelem ovládacího panelu. Tento typ ovládacího panelu umožňuje zobrazování a editaci hodnot proměnných a ovládaní aplikace systémem menu.

Druhý podporovaný typ panelu je prostá obrazovka. Nazýváme ho zjednodušený model ovládacího panelu. Jedná se o implementaci zobrazovacího prvku typu obrazovka jako hlavního a jediného kontaktu aplikace s uživatelem. Tento typ ovládacího panelu je vhodný pro použití návrhu obrazovek a zobrazení pro aplikace ovládané jinými postupy, než předpokládá programovací nástroj PESMENU. Má usnadnit tvorbu zobrazení hodnot a informací na displeji automatu.

O volbě druhu ovládacího panelu rozhoduje první vložený prvek editoru menu. Pokud zvolíme prvek menu nebo nadpis menu, je zvolen úplný model ovládacího panelu. V případě, že vložíme prvek obrazovka, zvolí PESMENU zjednodušený model ovládacího panelu. Zjednodušený model není možné dále strukturovat a tvoří ho právě jeden zobrazovací prvek typu obrazovka.

Vkládání nových prvků do struktury menu

Nabídka pro vládání prvků struktury menu vyvoláme z editoru menu stiskem klávesy Ins nebo stiskem kombinace kláves Shift-Ins. V prvním případě vkládáme položky menu a v druhém případě vkládáme libovolný ovládací prvek menu. Úkon vkládání je uvedeným způsobem rozdělen proto, aby byla snazší tvorba základního stromu struktury menu, tj. tvorba jednotlivých položek menu a jejich návazností. Definice výkonných akcí pro každou položku menu pak provádíme vkládáním dalších ovládacích prvků.



Obr. 4-9 Vkládání libovolného prvku

Po stisku klávesy Ins se vloží položka menu buď automaticky a nebo se zobrazí dialogové okno s výpisem typů položek, které je možné vložit. Automatické vložení proběhne tehdy, je-li možné vkládaný typ jednoznačně určit, tj. není-li volba vkládaného typu možná. Dialogová okna pro vkládání prvků menu do struktury menu jsou uvedena na Obr. 4-9 a Obr. 4-10.



Obr. 4-10 Vkládání položek typu menu a nadpis menu

4.3. Editor zobrazení ovládacích prvků menu

Editor zobrazení ovládacích prvků menu se skládá z pracovní plochy displeje a dvou tlačítek Ano a Zruš. Pracovní plocha displeje má tři zobrazovací roviny, odpovídající rovině podkladu, rovině textů a rovině dat.

V rovině podkladu můžeme kreslit za pomoci myši podklad zobrazení. Kreslení provádíme tak, že stiskem pravého tlačítka při současném umístění kurzoru do pracovní plochy displeje vyvoláme dialogové okno pro výběr kreslicího nástroje. Kreslicím nástrojem může být libovolný znak z dostupné znakové sady.



Obr. 4-11 Editor zobrazení

Kreslicí nástroj vybereme tak, že umístíme kurzor myši na požadovaný znak a stiskneme levé tlačítko myši. Poté potvrdíme takto vybraný znak stiskem pravého tlačítka. Dialogové okno pro výběr znaku se uzavře. Umístíme kurzor myši na vybranou pozici pracovní plochy displeje. Stiskneme levé tlačítko a pohybuje myši pořad vodorovně nebo svisle. Za kurzorem myši je kreslena vodorovná nebo svislá čára, složená z vybraného kreslicího nástroje (znaku). Chceme-li vybraný znak umístit jednotlivě, pak posuneme kurzor myši na požadovanou pozici a stiskneme levé tlačítko dvojstiskem.

Vkládání textů a zobrazovacích prvků do příslušných zobrazovacích rovin vyvoláme stiskem klávesy Ins při aktivní pracovní ploše displeje. Vyvoláme tak nabídku se všemi dostupnými typy prvků zobrazení. K dispozici jsou položky:

- TEXT** vloží textové pole do zobrazení.
- OBECNÝ** vloží odkazové pole na uživatelský typ zobrazovacího prvku.
- VÝČET** vloží odkazové pole na prvek typu výčet.
- HODNOTA** vloží odkazové pole na prvek typu hodnota.

Po vložení vybraného prvku můžeme pomocí myši prvek umístit a nastavit jeho rozměr. Umístění prvku provedeme pomocí myši. Kurzor myši přesuneme přibližně do poloviny pole prvku. Stiskneme levé tlačítko a pohybem myši umístíme pole prvku na požadovanou pozici. Tlačítko uvolníme.

Změnu velikosti zobrazovaného pole provedeme obdobně. Rozdíl bude pouze v tom, že před stiskem tlačítka a tažením myši umístíme kurzor myši na poslední znak pole. Pole odstraníme tak, že umístíme kurzor myši na první znak pole a stiskneme levé tlačítko myši.

Máme-li umístěno více zobrazovacích prvků v rovině textů a dat, vybíráme prvek k editaci tak, že umístíme kurzor myši doprostřed pole prvku a stiskneme levé tlačítko. Ke každému takto umístěnému prvku existuje dialogové okno pro nastavení parametrů a každý prvek má nastaveny implicitní parametry, které je však nutné modifikovat. Vyvolání dialogového okna pro modifikaci parametrů u vybraného prvku provedeme stiskem pravého tlačítka myši při současném umístění kurzoru myši na pole prvku.

Je nutné se ještě zmínit o prvcích zobrazení, jejichž existenci v editoru zobrazení řídí programovací nástroj PESMENU. **PESMENU vkládá tyto automatické prvky pouze ve dvou případech. Jedná se o případ editoru výčtu a editoru hodnoty. V těchto případech je na ploše displeje vloženo pole příslušného editoru. Toto pole nelze odstranit a ani nelze přidat další.** Tím je zajištěno, že pro ovládací prvek editor bude vždy k dispozici pole pro zobrazení editované hodnoty. U těchto speciálních prvků editoru zobrazení je nutné definovat parametry editoru a je možné je umísťovat podle uvážení po ploše displeje. Přemísťování polí editorů a definice jejich parametrů probíhá stejně jako u kteréhokoli jiného zobrazovacího prvku.

Všechny ostatní prvky z roviny dat jsou typu odkazu. U odkazů není nutné definovat všechny parametry, pokud k těmto odkazům existují editory kdekoli v struktuře menu. Pro jednotlivé zobrazovací prvky je nutné nastavit následující parametry:

- TEXT** v textovém poli provedeme editaci pouze jeho obsahu.
- OBECNÝ** poli pro obecný (uživatelský) zobrazovací prvek přiřadíme pouze jedinečné jméno, tj. název podle něhož bude PESMENU prvek identifikovat.
- VÝČET** pole výčet vyžaduje přiřazení jména a zadání textů přiřazených jednotlivým hodnotám výčtového typu. Textů může být maximálně 10 a musí být zadány postupně. Prvek výčtu pomocí textů zobrazuje vždy spojitou řadu hodnot (bez vynechání). Počtem vyplněných textových položek se řídí datový rozsah výčtu. V případě, že k výčtu existuje kdekoli v struktuře editor výčtu stejného jména, pak textové položky nevyplňujeme - budou převzaty z editoru. Pokud je přesto vyplníme, budou v průběhu generování výstupního zdrojového textu nahrazeny položkami, definovanými ve stejnojmenném editoru.

- **HODNOTA** pole hodnota je zobrazovací pole pro zobrazení proměnných v číselném formátu. V tomto poli musíme zadat jméno, zvolit formát zobrazovaného čísla a zarovnání výstupu čísla na displej. Minimální a maximální hodnotu musíme zadat pouze u automatického pole editoru hodnoty. V případě, že ji zadáme u zobrazovacího pole hodnoty (odkaz) a k tomuto poli bude existovat ve struktuře menu editor stejného jména, převezmou se údaje o maximu a minimu z něj.

Pozn. Pokud použijeme ve struktuře menu více editorů stejného typu a jména, jsou tyto editory ztotožněny.

Dialogová okna pro editaci parametrů

Jak již bylo uvedeno výše, jsou dialogová okna pro editaci parametrů určena pro zadávání specifických vlastností zobrazovacích a editačních prvků. Všechna tato okna mají společnou charakteristiku v tom, že editační řádky textů umožňují vkládat uživatelsky definované znaky.

Uživatelsky definované znaky jsou znaky, jejichž tvar si uživatel volí podle vlastního uvážení a vytváří si tak vlastní znakovou sadu, která je specifická právě těmito volitelnými znaky (blíže kapitola Editor znakové sady). Řádkové editory textů, použité v dialogových oknech, mají základní vlastnosti editorů. Máme na mysli jejich ovládání. Mimo toho, že zobrazují a editují základní znaky z ASCII tabulky (těmito znaky jsou vybaveny displeje automatů systému PES), zobrazují a editují také uživatelsky definované znaky. Těchto znaků může uživatel definovat 8.

Editor vloží zobrazení uživatelského znaku v případě, že zadáme jeho číselný kód. Zmíněný kód zadáme tak, že nastavíme kurzor editoru na místo, kam chceme vložit odpovídající uživatelský znak. Po té stiskneme klávesu Alt společně s číselným kódem znaku. V popisovaném případě jsou akceptovány kódy od 0 do 7. Editor na místo kurzoru vloží znak odpovídajícího kódu.

Dialogové okno pro editaci prvku TEXT a OBECNÝ

Na Obr. 4-12 je uvedeno dialogové okno prvku TEXT a prvku OBECNÝ (uživatelský zobrazovací prvek). Pomocí řádkového textového editoru zapíšeme formát textu



Obr. 4-12 Dialogové okno pro editaci prvku TEXT a OBECNÝ

pro prvek TEXT nebo zadáme jedinečné jméno pro prvek OBECNÝ. Editační změny uložíme stiskem tlačítka Ano. Dialogové okno uzavřeme stiskem tlačítka Zruš v případě, že nechceme, aby byly editační změny akceptovány.

Dialogové okno prvku VÝČET a EDITOR VÝČTU

Na Obr. 4-14 je uvedeno dialogové okno pro editaci parametrů prvku VÝČET a prvku EDITOR VÝČTU. Okno je vybaveno editorem pro zadání jména prvku a deseti editory pro přiřazení zobrazovaných textů hodnotám proměnné.



Obr. 4-13 Dialogové okno pro prvky HODNOTA a EDITOR HODNOTY

Akceptování editačních změn potvrdíme stiskem tlačítka Ano. Zrušení editačních změn volíme stiskem tlačítka Zruš. Čísla vlevo od editorů položek odpovídají přiřazené hodnotě proměnné pro editovaný text.



Obr. 4-14 Dialogové okno prvků VÝČET a EDITOR VÝČTU

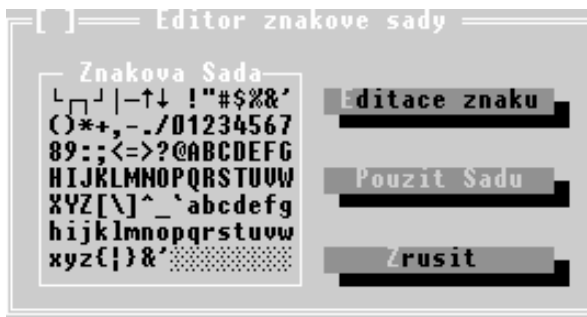
Dialogové okno prvku HODNOTA a EDITOR HODNOTY

Dialogové okno na Obr. 4-13 prvků HODNOTA a EDITOR HODNOTY je vybaveno řádkovým editorem jména, soupiskou použitelných formátů číselných výstupů, editory pro zadání maximální a minimální hodnoty a přepínačem pro formát zarovnání číselného výstupu včetně zobrazování ve znaménkovém formátu. Formát číselného výstupu volíme

pomocí kurzoru v soupisce formátů. Editační změny potvrdíme tlačítkem Ano. Zrušení editačních změn volíme stiskem tlačítka Zruš.

4.4. Editor znakové sady

Vzhledem k tomu, že programovatelné automaty systému PES jsou vybaveny textovými displeji s možností definovat až osm vlastních (uživatelských) znaků, je programovací nástroj PESMENU vybaven editorem znakové sady, aby bylo možné plně využít vlastností těchto displejů. Znaková sada může být různá pro různé návrhy menu a proto je vždy součástí souboru editoru menu. Tím je zajištěna nezávislost návrhů menu na počítači, na němž je menu aktuálně zpracováváno.



Obr. 4-15 Editor znakové sady

Problém uživatelských znaků ve znakové sadě je poněkud komplikovanější, než by se na první pohled zdálo. Jedná se o to, jakým způsobem zobrazovat uživatelské znaky v textovém okně editoru menu. Používá se pro to poměrně jednoduché řešení. Uživatel si nadefinuje tvar vlastního znaku bitovou mapou a poté takto definovanému znaku přiřadí znak z ASCII tabulky. Uživatelský znak bude pak na obrazovku vypisován v podobě přiřazeného znaku z ASCII tabulky.

Vzhledem k tomu, že se uživatelské znaky implementují obvykle jako znaky české abecedy nebo jako znaky rámečků či šipek, najdeme obvykle v tabulce ASCII vhodného reprezentanta.

Na Obr. 4-15 je uveden editor znakové sady, tvořený dialogovým oknem s výpisem sady a třemi tlačítky. Výpis znakové sady slouží pro orientační zobrazení použitých znaků a k výběru uživatelského znaku pro úpravu. Upravovat je možné pouze první osm znaků. Ostatní jednak nejsou volitelné a jednak odpovídají znakům zobrazovaným na displeji automatů systému PES.

Na obrázku Obr. 4-15 v poli znaková sada je zobrazena výchozí znaková sada programovacího nástroje PESMENU. V této standardní sadě jsou definovány uživatelské znaky rámečků a svislých šipek. Stiskem tlačítka Editace znaku vyvoláme editor bitové mapy znaku. Tlačítko Použit sadu slouží k ukončení editace znakové sady s následným zavedením této sady do editovaného souboru menu. Pokud obsahují texty souboru, jehož znakovou sadu měníme, uživatelské znaky z předchozí sady, jsou tyto znaky nahrazeny znaky novými v pořadí v němž byly uvedeny v původní znakové sadě.

Na Obr. 4-16 je uveden editor bitové mapy používaný pro editaci vzhledu zvoleného uživatelského znaku. V editoru bitové mapy označujeme stiskem mezerníku pozice bodů znaku které mají při zobrazení znaku svítit. Pokud je tato pozice již označena, nové stisknutí mezerníku provede zrušení označení. V poli Znak zobrazovat kódem pak volíme pozici kurzoru kód ASCII tabulky, který bude sloužit k zobrazování znaku na obrazovce editoru menu. Editační změny bitové mapy znaku můžeme buď potvrdit stiskem tlačítka Použít a nebo zrušit (tlačítko Zrušit).



Obr. 4-16 Editor bitové mapy znaku

4.5. Nastavení prostředí PESMENU

Některé vlastnosti prostředí programovacího nástroje je možné nastavit podle přání uživatele. Slouží k tomu dialogové okno Nastavení Prostředí (Obr. 4-17). Pomocí položek v tomto oknu můžeme volit rozměr obrazovky prostředí, automatické ukládání souborů zdrojových textů editoru menu a pracovní plochy, můžeme volit typovou řadu, pro niž bude výsledný kód generován a příponu výstupních souborů.

V příkladu je uvedena přípona STP používaná pro zdrojové texty jazyka SIMPLE. Zvolené nastavení prostředí můžeme potvrdit stiskem tlačítka Ulož nebo změny nepotvrdíme, stiskneme-li tlačítko Zruš.



Další nastavovací prvkem prostředí PESMENU je nastavení barev jednotlivých zobrazovaných prvků prostředí. Zmíněný dialog umožňuje též zvolit znak, který se tiskne na podklad pracovní plochy. Dialog nastavení barev vyvoláme příkazem Barvy z položky Pomocné hlavního Menu.

Obr. 4-17 Dialog pro nastavení prostředí PESMENU

5. ŘEŠENÉ ÚLOHY OVLÁDACÍCH PANELŮ

Další odstavce tohoto manuálu jsou věnovány zevrubnému popisu dvou řešených příkladů. Oba jsou navrženy pro automaty systému PES řady K. První příklad je řešen ve zjednodušeném modelu ovládacího panelu, druhý pak v modelu úplném.

5.1. Ručkové měřidlo na displeji automatu řady PES-K

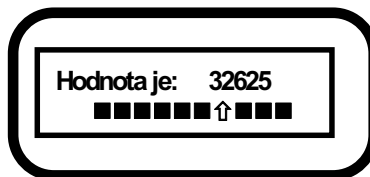
Zadání: Zobrazte hodnotu proměnné typu word na displeji automatu ve formátu simulující ručkové měřidlo společně s číselnou hodnotou.

Řešení:

V zadání příkladu je požadováno zobrazit hodnotu proměnné v číselném rozsahu 0 do 65535 jednak v číselném formátu a jednak v podobě analogového měřidla. Vzhledem k tomu, že v příkladu není požadováno ovládání aplikace a není požadována editace hodnoty, zvolíme pro řešení zjednodušený model ovládacího panelu. Ovládací panel bude tedy tvořen pouze zobrazovacím prvkem typu *Obrazovka*.

❑ Krok 1 - Návrh zobrazení

V prvním kroku řešení navrheme zobrazení, tj. tvar výpisu na obrazovku. Výpis na obrazovku použijeme podle (Obr. 5-1). Silné zarámování se zaoblenými rohy naznačuje displej automatu. Na čtyřech řádcích vytiskneme v rovině podkladu rámeček a v rovině textů přidáme text „Hodnota je:“.



Obr. 5-1 Návrh zobrazení hodnoty

V rovině dat pak přidáme zobrazovací prvek odkaz na hodnotu a současně přidáme zobrazovací prvek odkaz na výčet. Použijeme standardní znakovou sadu u níž provedeme úpravu tak, že místo šipky dolů budeme definovat plný znak pro výpis odkazu na výčet. (viz. obrázek).

❑ Krok 2 - Založení souboru menu a editace znakové sady

V druhém kroku spustíme programovací nástroj PESMENU. Z položky menu *SOUBOR* vybereme položku *NOVÝ*. Na pracovní ploše se objeví editor menu s vloženou základní položkou *Aplikace*.

Nejprve provedeme změnu znakové sady. Z menu vybereme příkaz *ZNAKOVÁ SADA* a nebo stiskneme kombinaci kláves *Alt-C*. Otevře se editor znakové sady. Myši

klepeme na pole výpisu znakové sady a přemístíme kurzor pod znak šipka dolů (↓). Myší stiskneme tlačítko Editace znaku a otevřeme editor uživatelského znaku. V editoru uživatelského znaku nejprve v poli Bitová mapa upravíme body znaku tak, aby byly vyplněny všechny řádky, vyjma řádku posledního. V poli „Znak zobrazovat kódem“ nastavíme kurzor na znak plného obdélníčku. Pokud jsme hotovi, stiskneme tlačítko Použít. Tím se vrátíme zpět do editoru znakové sady a místo znaku šipka dolů uvidíme zobrazen znak plného obdélníčku. Stiskneme tlačítko Použít a vrátíme se do editoru menu.

V tomto okamžiku máme všechny počáteční změny provedeny a uložíme je do souboru na pevný disk. Zvolíme položku ULOŽIT z menu SOUBOR. V dialogovém okně vyplníme název souboru na PR1.MNU a stiskneme tlačítko Uložit. Z ikony okna editoru menu zmizí hvězdička, čímž dává editor najevo, že všechny provedené změny uložil.

❑ Krok 3 - Volba zjednodušeného modelu ovládacího panelu

Jak již bylo uvedeno výše, volbu modelu provádí automaticky prostředí PESMENU podle první vložené položky. Stiskneme tedy v editoru menu tlačítko Ins a objeví se dialogové okno s nabídkou ovládacích prvků (Obr. 5-2), které je možné vložit. Přepínačem Objekt zvolíme Displej a stiskneme tlačítko Vložit.



Obr. 5-2 Nabídka vkládaných prvků

Na popisovanou volbu zareaguje editor menu vložením zobrazovacího prvku typu obrazovka Displej a aktualizuje výpis v okně dle Obr. 5-3. Nyní můžeme stisknout opět tlačítko Ins. Vyvolá se opět nabídka z Obr. 5-2. Pokud zvolíme opět Displej nestane se nic, pokud zvolíme Položku nebo Linku dojde k vymazání původně vloženého prvku Displej a místo něj se vloží prvek zvolený. Tímto mechanismem se vlastně volí modely ovládacího panelu.



Obr. 5-3 Výpis editoru menu

Jak je vidět, tak zjednodušený model je založen vždy na prvku Displej, model úplný pak na prvcích Menu tj. Položka nebo Linka. V našem příkladě jsme tedy zvolili zjednodušený model.

❑ Krok 4 - Úprava prvku Displej

V tomto kroku nás čeká modifikace prvku Displej podle návrhu zobrazení, uvedeného na Obr. 5-1. Stiskneme šipku dolů a přemístíme tak řádkový kurzor editoru menu na položku Displej. Stiskneme kombinaci kláves Ctrl-Enter a vyvoláme editor zobrazení (Obr. 5-4). Pomocí editoru zobrazení budeme nejprve editovat rovinu podkladu. Do této roviny umístíme rámeček z návrhu zobrazení.

Umístění rámečku do roviny podkladu

Stiskneme pravé tlačítko myši a ze zobrazení znakové sady vybereme znak vodorovné linky. Uzavřeme výběr kreslicího nástroje stiskem pravého tlačítka myši. Umístíme kurzor myši na první řádek displeje a za současného stisku levého tlačítka myši a tažení kurzoru nakreslíme horní vodorovnou čáru. Na konci čáry pustíme levé tlačítko myši. Obdobně nakreslíme spodní čáru rámečku. Pak zvolíme znak svislé čáry a nakreslíme levou a pravou stranu rámečku.



Obr. 5-4 Editor zobrazení

Jsme-li hotovi, zvolíme znak levého horního rohu a kurzor myši umístíme na první znak obrazovky displeje. Na levé tlačítko myši aplikujeme dvojitisk a umístíme tak jednotlivý znak rohu. Obdobně zvolíme a umístíme znaky zbývajících rohů. Popsaným způsobem bychom mohli vypsát i nápis „Hodnota je:“. Postup takového výpisu by byl zdlouhavý a proto pro vypsání nápisu použijeme rovinu textů.

Vypsání textu v rovině textů

Stiskneme klávesu *Ins* a z vyvolané nabídky vybereme položku *TEXT*. V levém horním rohu plochy displeje se objeví prázdné pole textu. Pomocí myši přemístíme toto pole na druhý řádek displeje a stiskneme pravé tlačítko myši. Stiskem tlačítka vyvoláme dialogové okno pro nastavení parametrů textu. Do řádkového editoru „Jméno“ vepíšeme text „Hodnota je:“ a stiskneme tlačítko *Ano*. Tím se zmodyfikuje obsah pole text. Pomocí myši teď upravíme velikost tohoto pole tak, aby byla zobrazena všechna písmena a práce v rovině textů je tak ukončena. Dále budeme postupovat v editaci roviny dat.

Editace roviny dat - vložení odkazu hodnoty a výčtu

Zbývá dvě pole navrhovaného zobrazení, tj. pole pro výpis číselné hodnoty a pole pro simulaci ručkového měřidla vložíme jako odkazy. V prvním případě jako odkaz na hodnoty a ve druhém případě jako odkaz na výčet. Stiskneme opět klávesu *Ins* a z nabídky vybereme položku *HODNOTA*. Do pravého horního displeje rohu se vloží pole odkazu na hodnotu. Toto pole pomocí myši umístíme na pravou stranu druhého řádku a upravíme jeho rozměr na pět znaků, abychom získali představu o umístění číselné hodnoty ve výpisu.

Stiskneme kombinaci kláves *Ctrl-Enter* a nastavíme parametry odkazu. Řádkovým textovým editorem vepíšeme jméno odkazu *Hodnota*. Z pole formátů vybereme formát základní formát „X“, tj. formát 0. Na závěr vybereme typ zarovnání „A“, tj. vlevo a pouze kladná čísla (neznaménkový výstup). Uzavřeme okno stiskem tlačítka *Ano* a vrátíme se tak do dialogového okna editoru zobrazení.

Obdobným způsobem vložíme a umístíme odkaz na výčet. Velikost pole tohoto odkazu upravíme na deset znaků. Stiskem kombinace kláves Ctrl-Enter vyvoláme editor pro nastavení parametrů odkazu na výčet. Pomocí rádkového editoru jména vložíme název odkazu „meric“.

Nyní budeme editovat jednotlivé texty položek. První položku vytvoříme tak, že na pozici prvního znaku umístíme znak šipky nahoru stiskem kombinace kláves Alt-6 (jedná se o uživatelský znak). Obdobným způsobem vepíšeme zbylých devět znaků (plný obdélníček) položky stiskem kombinace kláves Alt-7. Nyní postoupíme k další položce. Výpis upravíme tak, aby byl znak šipky umístěn na pozici druhého znaku položky a všechny ostatní znaky byly plné obdélníčky. Analogickým postupem vyplníme zbylé položky výčtu. Jsme-li, hotovi potvrdíme nastavení stiskem tlačítka Ano. Ze zápisu parametrů je již patrné, jakým způsobem bude simulováno ručkové měřidlo.

□ Krok 5 - Generování výstupního zdrojového textu

Poslední krok, který v programovacím nástroji PSMENU uděláme, bude generování výstupního kódu, tj. zdrojového textu v jazyce Simple. Generování kódu vyvoláme buď příkazem *PŘEKLAD* nebo stiskem kombinace kláves Alt-F9. Po dokončení procesu generování uzavřeme PSMENU a v prostředí PESPRO otevřeme výsledný zdrojový text návrhového prostředí. V PSMENU provedeme nezbytné doplnění textu o ovládání programového rozhraní.

□ Krok 6 - Získaný zdrojový text a jeho úprava

Jak je vidět, generátor kódu vygeneroval poměrně rozsáhlý zdrojový text, jehož strukturu se pokusíme vysvětlit.

Úvodní definice, deklarace a společné podprogramy

Soubor zdrojového textu PR1.STP začíná úvodními definicemi a deklaracemi. Nejprve jsou definovány konstanty jednotlivých kláves klávesnice, následují definice konstant pro proměnnou **G_Init**. Proměnná **G_Init** v průběhu obsluhy ovládacího panelu obsahuje stav zobrazení, v němž se zvolený ovládací prvek nachází. Dále následuje deklarace proměnných, které ovládací panel využívá. Všimněme si, že ovládací panel zabírá proměnné **D0** až **D13**. Na první pohled se to zdá hodně, nicméně jak se ukáže dál, tak proměnné **D0-D11** využívá pouze v době, když jsou podprogramy panelu vykonávány, po zbytek doby (programu) jsou proměnné včetně jejich hodnot vráceny aplikaci. Proměnné **D12** a **D13** jsou vyhrazeny pro programové rozhraní předávání dat a aplikace je musí využívat předepsaným způsobem (viz. Aplikace a její ovládání str. 5).

Ovládací panel má všechna potřebná data a datové struktury uloženy v zásobníku. Tomu odpovídají i další řádky zdrojového textu, na nichž najdeme deklaraci bázevých adres zásobníku a deklarace jednotlivých datových struktur. Po úvodních deklaracích již následují jednotlivé podprogramy ovládacího panelu.

Nejprve jsou to podprogramy pro zavedení znakové sady do ovladače displeje (podprogramy *CharSet0-CharSet7*). Následují dva podprogramy *G_DSave* a *GD_Load* s inverzní funkcí. Podprogramy zajišťují ukládání datových proměnných **D0-D11** do zásobníku před voláním podprogramů ovládacího panelu a obnovu hodnot těchto proměnných po ukončení volání podprogramů ovládacího panelu. Z uvedeného vysvětlení je tedy zřejmé, že proměnné **D0-D11** může mimo podprogramy ovládacího panelu využívat i aplikace k uschování a zpracování hodnot.

Dalším podprogramem ovládacího panelu je podprogram *Init_Menu*, obsahující inicializační procedury datových struktur ovládacího panelu. Zde můžeme měnit jednotlivé počáteční hodnoty použitých proměnných.

Dále následují v textu podprogramy *ClrScr*, *ClrLine*, *ClrCur*, obsluhující různé způsoby mazání displeje. Následují podprogramy pro obsluhu klávesnice (*GlobalUp*, *GlobalDown*, *GlobalDownH*, *GlobalCode*, *GlobalRol* a *GlobalRolH*). Po těchto úvodních programových řádcích následují podprogramy jednotlivých zobrazovacích prvků.

Podprogramy zobrazovacích prvků

Podprogramy jednotlivých ovládacích a zobrazovacích prvků jsou generovány v pořadí stromových struktur, užitých k ukládání dat v programovacím nástroji PESMENU. Pořadí podprogramů se může měnit v závislosti na postupu editace a návrhu ovládacího panelu v PESMENU.

M2Show podprogram pro zobrazení odkazu na hodnotu. Struktura podprogramu neobsahuje nastavení pozice výpisu odkazu. Za nastavení pozice je v tomto případě vždy zodpovědný podprogram, který výpis odkazu volá.

MRef100Char až

MRef199Char podprogramy pro výpis jednotlivých modifikací odkazu na výčet. Poslední číslice v názvu odpovídá číslu položky z editoru parametrů prvku výčet. Vzhledem k tomu, že tyto položky byly konstruovány z uživatelských znaků, je potřeba vygenerovat speciální typ výstupu těchto řetězců na displej. To zajišťují právě podprogramy těchto jmen.

M1Show podprogram je řídicím podprogramem zobrazení proměnné typu odkaz na výčet. Všimněme si, že v závislosti na hodnotě datové proměnné **Data1** volá jednotlivé podprogramy ze sady **MRef100Char** až **MRef199Char**.

M01Char až

- M03Char** podprogram generovaný pro výtisk rámečku v rovině textu a podkladu základního prvku Displej. Tyto podprogramy jsou generovány z důvodu použití uživatelských znaků v rovině podkladu.
- MODisplay** základní zobrazovací podprogram pro zobrazovací prvek Displej. Povšimněme si u tohoto podprogramu poněkud blíže vygenerovaného kódu. V prvních čtyřech řádcích se provádí tisk na displej v rovině podkladu a textů. Vzhledem k tomu, že byly použity uživatelské znaky je výtisk těchto rovin na displej automatu poněkud komplikován voláním podprogramů pro tisk rámečku. V posledním řádku je pak generován text na tisk odkazů v rovině dat. Zde je dokumentován předpoklad, že za nastavení pozice pro zobrazovací podprogramy je odpovědný vždy podprogram volající tato zobrazení. Je to tak proto, aby se při každém použití odkazu daného jména (odkaz v různých částech struktury menu) nemusel generovat zvláštní podprogram, který by se odlišoval pouze jiným nastavením proměnné **POSITION**.
- MOMain** řídicí podprogram pro zobrazování prvku „Displej“. Podprogram reaguje na stav proměnné **G_Init** a podle tohoto stavu spouští zobrazení jen tehdy, pokud je to nutné, tj. pokud je o to požádám nastavením proměnné **G_Init** na hodnotu *Redraw*.
- MainMenu** je hlavní podprogram ovládacího panelu. Zde je nutné vysvětlit jednotlivé programové řádky. V prvním řádku probíhá uschování dat aplikace z proměnných **D0-D11** a inicializace těchto proměnných dat ovládacího panelu. Inverzní proces je řešen v řádku posledním. Druhý, třetí a čtvrtý a pátý řádek zajišťují ovládání datové struktury programového rozhraní mezi panelem a aplikací. Druhý řádek iniciuje překreslení displeje v případě, že je přijata zpráva z aplikace. Třetí a čtvrtý řádek aktualizují při přijetí odpovídající zprávy datové proměnné obou vložených odkazů (odkaz na výčet - *meric*, odkaz na hodnotu - *hodnota*). Pátý řádek pak zajišťuje smazání proměnné **G_Message** pro informaci, že došlo ke zpracování zprávy. Šestý řádek volá řídicí funkci zobrazovacího prvku „Displej“. Sedmý řádek zajišťuje inicializaci řídicí proměnné v případě přetečení její maximální hodnoty. Jedná se o bezpečnostní řádek. Když ho vynecháme a proměnná **G_Item** přeteče dovolený rozsah, přestane ovládací panel fungovat.

Hlavní programová smyčka

Hlavní programová smyčka má v našem případě pouze dva řádky. První je inicializační. Zde se volá inicializační podprogram **Init_Menu** a maže se příznakový bit **RESET**. V dalším řádku se již volá pouze hlavní podprogram ovládacího panelu **MainMe-**

nu. Nyní provedeme jednoduchou úpravu kódu hlavní programové smyčky, kterou budeme simulovat kód aplikace. Před inicializační řádek vložíme následující text:

```
IF RESET THEN Y30 : D20 = 0
```

Tímto řádkem provedeme rozsvícení podsvětlení displeje a inicializaci aplikační proměnné **D20**. Inicializační řádek ovládacího panelu se o smazání proměnné **RESET** postará, takže i tento řádek proběhne pouze jednou. Před volání hlavního podprogramu ovládacího panelu MainMenu vložíme následující řádek :

```
IF (KBCODE = KB_F1) THEN D20 = D20 + 7968 :
    G_MESSAGE = UN_WRITE+UN_2 :
    G_MESSDATA = D20
```

Tento programový řádek provede vždy po stisku klávesy F1 zvýšení proměnné D20 o číslo 7968. Dále provede naplnění hodnoty proměnné **G Message** a proměnné **G MessData**. Tím formátuje zprávu pro programové rozhraní. Program nyní přeložte, zatáhněte do programu a stiskem klávesy F1 vyzkoušejte jeho funkci. Při zkoušení zjistíte, že se při stisku klávesy F1 mění pouze výpis odkazu na hodnotu. Je to pochopitelné protože přes rozhraní šla pouze jedna zpráva.

Abychom ovládli i proměnnou odkazu výčtu musíme vyslat zprávy dvě. **Problém však spočívá v tom, že programové rozhraní umí v jednom běhu programové smyčky zpracovat pouze jednu zprávu.** Tento problém je možné řešit dvěma postupy. První postup je napsat kód aplikace tak, aby tomuto požadavku vyhověl. Druhá metoda spočívá v tom, že změníme hodnoty proměnných přímo v datových strukturách ovládacího panelu. Tento způsob však naráží na to, že ovládací panel o této změně neví a tudíž na ní nereaguje. Proto musíme vždy po takovém zásahu nastavit proměnnou **G Message** na hodnotu různou od nuly nejlépe na hodnotu *Překresli*. Ukažme si řešení:

```
IF (KBCODE = KB_F1) THEN D20 = D20 + 7968 :
    G_MESSAGE = UN_WRITE :
    POINTER = DATA2 :
    STACK = D20 :
    POINTER = DATA1 : STACK = D20 / 7000
```

Uvedeným zdrojovým textem nahradte řádek doplněný v předchozím kroku. Program přeložte a ověřte reakci výstupu na displej v případě stisku klávesy F1.

Pozn. Uvedené zdrojové texty úprav musí být v souboru zdrojového textu umístěny na jednom řádku. Protože však jsou zápisy textů dlouhé a na šířku řádku této publikace by se nevešly, použili jsme pro zápis více řádků zarovnaných vpravo.

5.2. Přepínač výstupů Y0 až Y7 a editor napětí na výstupu O0

Zadání:

Navrhněte ovládací panel automatu PES-K10, realizující přepínač výstupů Y0, Y1 a umožňující nastavit hodnotu napětí na analogovém výstupu O0. Obě funkce budou ovládány z klávesnice automatu.

Řešení:

Ze zadání vyplývá požadavek na dva editační prvky. Řešení nezávislého přepínače výstupů vede na použití editoru výčtu. Nastavení hodnoty výstupního napětí na analogovém výstupu vede na použití editoru hodnoty. Vzhledem k tomu, že jsou požadovány dva nezávislé editační prvky, musíme umožnit uživateli aplikace volbu mezi nimi. To zajistíme pomocí menu, které bude mít dvě položky. Jednu pro spuštění editoru hodnoty a druhou pro spuštění editoru výčtu. Aby bylo možné i při zobrazení menu podat informaci o nastavených hodnotách, doplníme tyto položky o odkazy na příslušné editory.

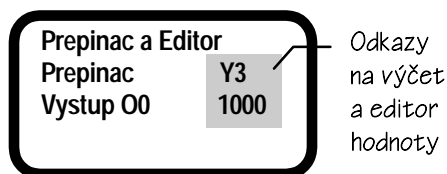
❑ Krok 1 - Návrh zobrazení

V přípravném kroku si provedeme náčrt, jak by měla všechna použitá zobrazení vypadat. Vzhledem k tomu, že výsledný kód generovaný generátorem PESMENU bude poněkud složitější než v předchozím případě, nebudeme používat uživatelské znaky.

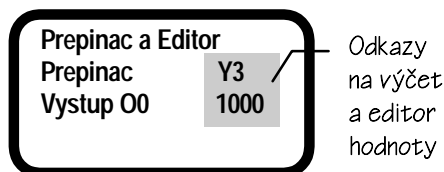
Pozn. Použití uživatelských znaků je samozřejmě možné kdykoliv, v našem příkladě se ho zřekneme pouze z důvodu kratšího a přehlednějšího výstupního souboru.

Začneme s návrhem hlavního menu podle Obr. 5-6. Spustíme programovací nástroj PESMENU a otevřeme nový soubor. Vzhledem k tomu, že nebudeme editovat znakovou sadu uložíme tento soubor pod názvem PR2.MNU.

Nyní v okně editoru menu stiskneme klávesu Ins a z nabídky vybereme položku **LINKA**. Vložením **LINKY** (nadpis menu) uděláme jednak vložení nadpisu menu, tj. realizujeme první řádek z Obr. 5-6 a současně zavedeme úplný model ovládacího panelu. Úplný model je zaveden vzhledem k tomu, že jsme jako první položku vložili položku typu menu (je jedno jestli to bude menu a nebo menu s nadpisem). Nyní stiskneme dvakrát za



Obr. 5-5 Návrh zobrazení menu



Obr. 5-6 Návrh zobrazení menu

sebou klávesu Ins. Všimněme si, že se editor menu neptá na typ položky, protože je předem známý. Nemůže to totiž být nic jiného než položka menu, neboť nadpis menu je již vložen (může být použit v dané úrovni menu pouze jednou a to pouze na prvním místě).



Obr. 5-7 Výpis editoru menu po editaci nadpisu

Nyní musíme upravit text menu tj. text nadpisu a text jednotlivých položek. Začneme s nadpisem. Umístíme kurzor editoru menu na položku „Nadpis Menu“, stiskneme kombinaci kláves Ctrl-Enter a vyvoláme tak editor zobrazení.

Všimněme si, že v tomto případě na rozdíl od předchozího příkladu má plocha displeje ve vswislem směru rozměr pouze jednoho řádku. To je proto, že nadpis menu se vypisuje právě na jeden řádek. Stiskneme klávesu Ins a vložíme Text. Editujeme pouze v rovině textů, protože podle návrhu je rovina podkladu prázdná a rovina dat neobsahuje žádný odkaz. Stiskneme kombinaci kláves Ctrl-Enter a pomocí řádkového editoru vložíme text „Prepínac a Editor“. Editaci ukončíme stiskem tlačítka Ano. Závěrečným úkonem bude umístění a úprava velikosti pole textu v editoru zobrazení. Jsme-li hotovi, stiskneme tlačítko Ano a ukončíme tak editaci nadpisu menu. Editor menu upraví po ukončení editace menu výpis dle Obr. 5-7.

Přesuneme kurzor editoru menu na první položku menu a opět stiskneme klávesy Ctrl-Enter abychom otevřeli editor zobrazení. Stiskneme klávesu Ins a vložíme pole textu. Stiskneme kombinaci kláves Ctrl-Enter a zapíšeme text položky menu „Prepínac“. Editaci zavřeme stiskem tlačítka Ano. Upravíme polohu a rozměr textového pole.

*Pozn. **IPOZOR!** Textové pole musíme umístit ve vzdálenosti alespoň jednoho znaku od levého okraje zobrazení. Na prvním znaku je zobrazován kurzor menu. Kdybychom umístitili textové pole hned od začátku, přepisoval by první znak textu znak tohoto kurzoru.*

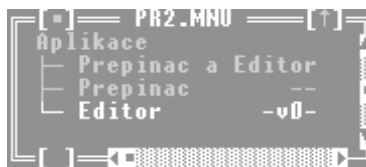
Stiskneme opět klávesu Ins. Z nabídky zvolíme tentokrát položku **VÝČET**. Tím do zobrazení vložíme prvek odkazu na výčet. Stiskneme kombinaci kláves Ctrl-Enter a otevřeme editor parametrů výčtu. Zde vyplníme pouze jméno výčtu „Prepínac“. Vzhledem k tomu, že k tomuto výčtu bude existovat editor výčtu, jednotlivé položky budou převzaty právě z definice tohoto editoru. Editaci parametrů ukončíme stiskem tlačítka Ano. Upravíme rozměr pole odkazu na dva znaky, neboť předpokládáme zobrazení pouze dvouznakové ve formátu Yx, kde x je číslo sepnutého výstupu. Editaci zobrazení ukončíme stiskem tlačítka Ano a editor menu upraví výpis podle Obr. 5-8. Všimněme si, že odkaz na výčet je zde zobrazen dvěma pomlčkami. To je



Obr. 5-8 Výpis editoru po editaci položky menu

proto, aby se odlišil od prostého textu. V případě, že bychom pole odkazu udělali víceznakové, zobrazovalo by se mezi pomlčkami jméno odkazu.

Posuneme kurzor editoru menu na poslední položku a stiskneme kombinaci kláves Ctrl-Enter, abychom vyvolali editor zobrazení. Vložíme textové pole „Editor“, upravíme rozměr pole. Stiskneme klávesu Ins a z nabídky vybereme položku **HODNOTA** a vložíme odkaz na editor hodnoty. Stiskneme kombinaci kláves Ctrl-Enter a vyvoláme editor parametrů odkazu na hodnotu. Vložíme jméno „v0“, ve výpisu formátů nastavíme kurzor na položku XXXX, typ výstupu a zarovnání zvolíme A (neznaménkově na čtyři znaky, zarovnat vpravo). Maximum a minimum nenastavujeme, protože budou přejata z odpovídajícího editoru. Stiskneme tlačítko Ano a ukončíme editaci parametrů. Nyní upravíme velikost pole na čtyři znaky a umístíme ho k pravému okraji zobrazení. Editor zobrazení uzavřeme stiskem tlačítka Ano. Editor menu upraví výpis menu do tvaru podle Obr. 5-9.

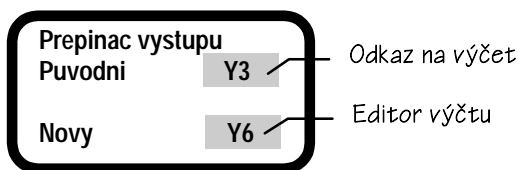


Obr. 5-9 Výpis editoru menu po celkové úpravě položek

Na výpisu vidíme nyní odkaz na editor a protože je nastavená velikost pole odkazu dostatečná, zobrazuje editor menu celé jméno odkazu mezi pomlčkami.

□ Krok 2 - Vložení editoru výčtu

V dalším kroku provedeme návrh zobrazení editoru výčtu, který použijeme pro realizaci přepínače výstupů. Budeme postupovat velmi obdobně jako při návrhu zobrazení v prvním kroku našeho příkladu. Nejprve si rozmyslíme jaký má mít tvar zobra-



Obr. 5-10 Návrh zobrazení editoru výčtu

zení tohoto editoru. Použijeme návrh z Obr. 5-10. Zobrazení budou tvořit texty „Prepinac vystupu“, „Puvodni“ a „Novy“. Tyto texty doplníme odkazem na výčet. Editor výčtu bude vložen automaticky programovacím nástrojem PSMENU.

Postup vytvoření editoru výčtu je jednoduchý. Přesuneme kurzor editoru na položku „Prepinac“ a stiskneme klávesu Enter. Tím se zobrazení editoru změní a my se vnoříme do další úrovně stromové struktury menu. Výpis bude nyní vypadat tak, že na prvním řádku bude vypsána položka „Prepinac“. Nyní můžeme stisknout klávesu Ins a z nabídky vybereme položku „Výčet“. Stiskneme tlačítko Vložit a ukončíme tak volbu prvku. Výpis editoru menu bude upraven tak, že na druhém řádku bude uvedena položka „Výčtový typ“. Pokusíme-li se o další stisk klávesy Ins nestane se nic, protože položka výčtového typu vždy ukončuje větev stromové struktury.

Přemístíme kurzor editoru menu na položku „Výčtový typ“ , stiskneme kombinaci kláves Ctrl-Enter a vyvoláme editor zobrazení. Všimněme si, že na ploše displeje je vloženo pole. Toto pole je pole editoru výčtu. Jak již bylo uvedeno výše, můžeme editovat pouze parametry tohoto pole. Není ho možné odstranit. Stiskneme kombinaci kláves Ctrl-Enter a otevřeme editor parametrů výčtového typu. Pomocí tohoto typu editoru jsme již editovali odkaz na výčet a je nám tedy již znám. V tomto případě však mimo jména „Prepinac“ vyplníme i jednotlivé texty položek výčtu. Použijeme texty se jmény výstupů Y0, Y1, Y2, Y3, Y4, Y5, Y6, Y7. Editor uzavřeme stiskem tlačítka Ano. Nyní upravíme velikost a umístění pole editoru podle Obr. 5-10.

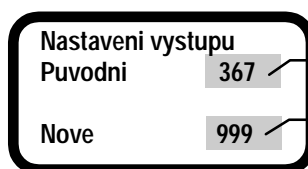
Stiskneme klávesu Ins a vložíme z nabídky položku VÝČET. Stiskneme kombinaci kláves Ctrl-Enter a otevřeme opět editor parametrů výčtu. Nyní postačí vyplnit pouze jméno „prepinac“. Tím, že použijeme identické jméno jako má automaticky vložený editor výčtového typu, zajistíme vzájemnou vazbu mezi odkazem a editorem. Vzhledem k tomu, že je jméno identické i s odkazem použitým v zobrazení položky menu, bude tato vazba nastavena i vzhledem k tomuto odkazu, který je v úplně odlišné části stromové struktury menu. Uzavřeme editor zobrazení a editor menu upraví výpis podle Obr. 5-11. Nyní posuneme kurzor editoru menu na první řádek a stiskneme klávesu Enter. Tím se vrátíme do úrovně hlavního menu.



Obr. 5-11 Výpis editoru menu po vložení editoru výčtu

❑ Krok 3 - Vložení editoru hodnoty

Z celého návrhu menu nám v tuto chvíli zbývá definovat zobrazení editoru hodnoty. Budeme postupovat analogicky ke kroku 2. Nejprve zvolíme kurzorem editoru menu položku „Editor“ a



Odkaz na hodnotu

Editor hodnoty

stiskneme klávesu Enter. Tím se dostaneme do další úrovně menu navrhovaného menu. Stiskneme klávesu Ins a z nabídky vybereme položku „Editor“. Tím vložíme editor hodnoty. Zobrazení editoru hodnoty upravíme podle návrhu na Obr. 5-12. Nezapomeneme zadat identická jména „v0“ editoru a odkazu. Odkaz vkládáme na rozdíl od předchozího kroku volbou HODNOTA. U odkazu a editoru nastavíme zarovnání doprava, zvolíme formát XXXX a u editoru navíc nastavíme položku maxima na hodnotu 1000 a položku minima na hodnotu 0. Vložíme taktéž textová pole dle Obr. 5-12. Ukončíme editaci zobrazení. Vrátime se do úrovně hlavního menu a návrh ovládacího panelu je hotov.

V tuto chvíli zbývá soubor uložit a spustit generátor menu známou kombinací kláves Alt-F9.

□ Krok 4 - Získaný zdrojový text a jeho úprava

Úvodní definice, deklarace a společné podprogramy

Soubor zdrojového text PR2.STP začíná úvodními definicemi a deklaracemi. Jejich řazení odpovídá formálně příkladu 1. Řada podprogramů je totožná. Vzhledem k tomu, že jsme použili editor hodnot, generátor přidal další společné podprogramy. Jsou určeny převážně na ovládání editoru hodnot. Jedná se o následující podprogramy:

- GlobalPForm** podprogram zajišťuje výpis chybového hlášení na displej, pokud se pokoušíme vložit hodnotu mimo zadanou toleranci a nebo pokud se k editaci hodnoty pokoušíme použít nepřipustný formát (použijeme-li např desetinnou tečku v celočíselném formátu).
- GlobalSetF** podprogram mění formát výpisu editoru v závislosti na stisknuté klávese tak, aby odpovídal zadanému formátu zobrazení a přitom aby v průběhu editace byla editovaná hodnota zobrazována obdobně jako na obyčejné kalkulačce.
- GlobalCheckF** podprogram pro kontrolu formátu editované položky spouštěný ve spolupráci s podprogramem **GlobalSetF**.
- GlobalCheckD** podprogram zajišťující kontrolu rozsahu editované hodnoty. V případě, že ukončíme editaci stiskem klávesy Ent na klávesnici automatu, volá se tento podprogram, aby zjistil zda je editovaná hodnota v zadaném rozsahu minima a maxima.
- GlobalEditL** podprogram zajišťuje editační funkce a je klíčovým pro činnost editoru hodnoty.
- GlobalRoundF** upraví editovanou hodnotu z virtuálního formátu, používaného při editaci do podoby, aby odpovídala zadanému formátu. Virtuální formát se mění v průběhu editace. Uvedme příklad. Představme si, že proměnná nabývá hodnotu z rozsahu 0 až 1000. Přitom se však při výstupu na displej chápe ve formátu X.XX, tj. na dvě desetinná místa. V případě, že stiskneme napřed 1, pak desetinnou tečku a nakonec 0 na klávesnici automatu, zobrazí se na displeji hodnota 1.0. To je správně, nicméně editor ji zobrazuje ve virtuálním formátu X.X. Proč to dělá? Důvod je prostý. Hodnota při editaci vzniká tak, že se vezme původní číslo v editované proměnné a znásobí se deseti. Poté se přičte číslo, odpovídající stisknuté klávese. Z uvedeného plyne, že v našem případě bude mít editovaná proměnná hodnotu 10 ač by měla mít hodnotu 100, aby byla zapsána v požadovaném formátu správně. Ukončíme-li v takovém okamžiku editaci, musí editor konečnou hodnotu v proměnné korigovat a k tomu slouží právě tento podprogram.

GlobalValidL jedná se o řídicí podprogram kontroly krajních mezí hodnoty proměnné při ukončení její editace.

Podprogramy zobrazovacích prvků

Podprogramy jednotlivých ovládacích a zobrazovacích prvků jsou generovány v pořadí stromových struktur, užitých k ukládání dat v programovacím nástroji PESMENU. Pořadí podprogramů se může měnit v závislosti na postupu editace a návrhu ovládacího panelu v PESMENU.

M2Show podprogram pro zobrazení odkazu na hodnotu, tj. odkaz na editor napětí výstupu **00**.

M1Show podprogram je řídicím podprogramem zobrazení proměnné typu odkaz na výčet, zobrazující sepnutý výstup **Y0-Y7**.

Dalšími podprogramy jsou řídicí podprogramy hlavního menu. Jedná se o:

M0Display podprogram pro zobrazení položek hlavního menu. Povšimněme si volání zobrazení odkazů. Z vygenerovaného textu je patrné, že pozice odkazů závisí na proměnné **G_Rol**. Pokud bude mít menu více položek, než má displej řádků, budou při procházení menu jeho jednotlivé položky rolovat. Rolování se zúčastní i jednotlivé odkazy, takže budou sledovat svoji položku menu.

M0Make podprogram zajišťuje vyvolání akce po té, co uživatel stiskl klávesu Enter na klávesnici automatu. Podle aktuální polohy kurzoru se řídí výběr volání akce.

M0Main řídicí podprogram pro zobrazování hlavního menu a volání podprogramu **M0Make** pro předání řízení k vybrané akci. V našem případě se předá řízení editoru výčtu a nebo editoru hodnoty.

Následují podprogramy editoru výčtu a editoru hodnoty. Všimněme si, že mají formálně stejnou strukturu jako podprogramy hlavního menu.

M2Display podprogram editoru výčtu, určený pro zobrazení navržené obrazovky editoru výčtového typu přepínač.

M2Make podprogram zajišťuje vyvolání akce po té, co uživatel stiskl klávesu Enter na klávesnici automatu. Formátuje se zpráva do proměnné **G_Message** a současně se aktualizuje hodnota výčtu.

M2Main řídicí podprogram editoru výčtu.

M1Display podprogram editoru hodnoty určený pro zobrazení navržené obrazovky editoru výstupního napětí v0.

- M1Make** podprogram zajišťuje vyvolání akce po té, co uživatel stiskl klávesu Enter na klávesnici automatu. Provádí se kontrola rozsahu, formátuje se zpráva do proměnné **G Message** a současně se aktualizuje hodnota editované proměnné.
- M1Main** řídící podprogram editoru hodnoty.

Hlavní programová smyčka

Hlavní programová smyčka má nyní čtyři řádky. První je inicializační. Zde se volá inicializační podprogram **Init_Menu** a maže se příznakový bit **RESET**. V dalším řádku se již volá pouze hlavní podprogram ovládacího panelu **MainMenu**. Za voláním hlavního podprogramu menu následují dva řádky pro zpracování zpráv, které generují editory menu. První z nich se týká přepínače a druhý editoru napětí na výstupu. Nyní provedeme jednoduchou úpravu kódu hlavní programové smyčky, kterou budeme simulovat kód aplikace. Před inicializační řádek vložíme následující text:

```
IF RESET THEN Y30 : Y0 : 00 = 0
```

Tímto řádkem provedeme rozsvícení podsvětlení displeje, sepnutí výstupu **Y0** a nastavení nulové hodnoty napětí na výstupu **00**. Inicializační řádek ovládacího panelu se o smazání proměnné **RESET** postará, takže i tento řádek proběhne pouze jednou. Nyní před hlavní programovou smyčku dopíšeme podprogram na zpracování zprávy z editoru výčtu. Podprogram bude vypadat takto:

```
SUBROUTINE ZAPNI
Y0' : Y1' : Y2' : Y3' : Y4' : Y5' : Y6' : Y7' ; VYPNUTI VYSTUPU
IF (G_MESSDATA = 0) THEN Y0
IF (G_MESSDATA = 1) THEN Y1
IF (G_MESSDATA = 2) THEN Y2
IF (G_MESSDATA = 3) THEN Y3
IF (G_MESSDATA = 4) THEN Y4
IF (G_MESSDATA = 5) THEN Y5
IF (G_MESSDATA = 6) THEN Y6
IF (G_MESSDATA = 7) THEN Y7
RETURN
```

Do programového řádku obsluhy zprávy z editoru výčtu vložíme volání uvedeného podprogramu. Řádek bude vypadat takto:

```
IF (G_MESSAGE = UN_2 ) THEN ZAPNI : G_MESSAGE = 0
```

Úprava řádku pro zpracování zprávy z editoru hodnoty napětí na výstupu **00** bude výrazně jednodušší. Do zdrojového textu řádku vložíme pouze přiřazovací příkaz. Řádek bude mít tvar:

```
IF (G_MESSAGE = UN_1 ) THEN O0 = G_MESSDATA : G_MESSAGE = 0
```

Takto upravený program přeložíme překladačem prostředí PESPRO, zatáh-
neme do automatu a ověříme jeho funkci. Sepnutý výstup můžeme sledovat na indikačních
diodách hodnoty napětí na výstupu můžeme měřit voltmetrem a nebo v prostředí PESPRO
můžeme použít okna vstupů a výstupů a nebo okna sledovače.

6. ZÁVĚREM

Uvedený popis programovacího nástroje PSMENU má za úkol provést uživatele tímto produktem a ozřejmit některé postupy při tvorbě aplikačních programů pro programovatelné automaty systému PES. Samotný programovací nástroj PSMENU má pomoci usnadnit a urychlit práci programátorů. Je zřejmé, že všechnu programátorskou dřinu převzít neumí a ani nemůže, nicméně pokud si na něj zvyknete, bude jistě Vaším dobrým pomocníkem.